3D Printing Designs: Fun and Functional Projects

Get to grips with 3D printing and learn to model designs using Blender

Joe Larson

Table of Contents

3D Printing Designs: Fun and Functional Projects

3D Printing Designs: Fun and Functional Projects

First published: June 2016

Production reference: 1210616

Credits

Author

Joe Larson

Reviewer

Marcus Ritland

Commissioning Editor

Edward Gordon

Acquisition Editor

Vinay Argekar

Content Development Editor

Shweta Pant

Technical Editor

Vishal K. Mewada

Copy Editor

Madhusudan Uchil

Project Coordinator

Kinjal Bari

Proofreader

Safis Editing

Indexer

Hemangini Bari

Graphics

Kirk D'Penha

Production Coordinator

Shantanu N. Zagade

Cover Work

Shantanu N. Zagade

About the Author

Joe Larson, known online as "the 3D Printing Professor," is one part artist, one part mathematician, one part teacher, and one part technologist. It all started in his youth, doing BASIC programming and low-resolution digital art on a Commodore 64. As technology progressed, so did Joe's dabbling, eventually taking him to 3D modeling while in high school and college, and he momentarily pursued a degree in computer animation. He abandoned that and instead became a math teacher, and then moved to software development for 10 years before returning to education, teaching technology in college.

When Joe first heard about 3D printing, it took root in his mind, and he went back to dust off his 3D modeling skills. In 2012, he won a Makerbot Replicator 3D printer in the Tinkercad/Makerbot Chess Challenge, with a chess set that assembles into a robot. Since then, his designs on Thingiverse have been featured on Thingiverse, Gizmodo, Shapeways, Makezine, and other places. He currently produces weekly videos about design for 3D printing on his YouTube channel, http://www.youtube.com/user/mrjoesays.

About the Reviewer

Marcus Ritland is a designer and 3D printing consultant in his small business, Denali 3D Design. Since 2008, he has been providing 3D modeling and 3D printing services as well as moderating the SketchUcation 3D printing forum (http://sketchucation.com/).

He has volunteered at a local makerspace, teaching SketchUp classes and leading 3D printing meetups. As an author of 3D Printing with SketchUp, he is currently on a quest to eliminate design-for-3D printing illiteracy.

www.PacktPub.com

eBooks, discount offers, and more

Did you know that Packt offers eBook versions of every book published, with PDF and ePub files available? You can upgrade to the eBook version at www.PacktPub.com and as a print book customer, you are entitled to a discount on the eBook copy. Get in touch with us at <customercare@packtpub.com> for more details.

At www.PacktPub.com, you can also read a collection of free technical articles, sign up for a range of free newsletters and receive exclusive discounts and offers on Packt books and eBooks.

https://www2.packtpub.com/books/subscription/packtlib

Do you need instant solutions to your IT questions? PacktLib is Packt's online digital book library. Here, you can search, access, and read Packt's entire library of books.

Why subscribe?

- Fully searchable across every book published by Packt
- Copy and paste, print, and bookmark content
- On demand and accessible via a web browser

Preface

3D printers have arrived! Complex and beautiful objects are available at the touch of a button in our schools, libraries, or even our homes. Whether or not you have a 3D printer, learning how they work and how to design for them is the best way to be a part of this new industrial age. And the best part is it doesn't cost a penny to learn to design for them.

This book will teach you the things you need to know about 3D printers. Then, you will use the robust and free software, Blender, to follow step-by-step instructions through several planned projects. You will gain the tools, techniques, and skills you need to make your own projects that you can print by yourself on a 3D printer near you and share with others online to print around the world.

What this book covers

Chapter 1, *3D Printing Basics*, will help you understand 3D printing basics, types of 3D printing, and how FFF printers work.

Chapter 2, *Beginning Blender*, will introduce Blender, how to set it up, and some basic and mid-level functionality. Knowing the content of this chapter will get you over Blender's infamous learning curve and provide the basic knowledge and reference necessary for following along with future projects.

Chapter 3, *The Octopus Pencil Holder*, building this simple project, an octopus pencil holder, will involve simple selection techniques and modification commands of basic shapes in Blender's Edit mode, and applying modifiers to soften and combine shapes. This technique alone can be used to make an unlimited number of cool things once mastered.

Chapter 4, *Measuring Basics*, mentions how it is very important that accurate measurements must be made when planning and applied to the modeling of a 3D object. In this chapter, we deal with different techniques of taking measurements: measuring with a ruler or calipers, the grid paper trace method, and 3D scanning.

Chapter 5, *An SD Card Holder Ring*, walks you through the process of making a cool 3D printed project—an SD card holder.

Chapter 6, *Sculpting the Face of the Sun*, teaches you how to use the sculpting tools in Blender to

transform a basic shape into a complicated design.

Chapter 7, *Cutting a 3D Jigsaw Puzzle*, further modifies the model created in the previous chapter by creating the puzzle pieces and then applying their shape that to the sculpted model model to create the individual pieces, combining two of Blender's powerful editing styles to create a finished product.

What you need for this book

Blender has these minimum system requirements:

- A 32-bit dual-core 2-GHz CPU with SSE2 support

- 2 GB of RAM

- A 24-bit 1280×768 display

- A mouse or trackpad

- OpenGL 2.1 compatible graphics with 512 MB of RAM

Who this book is for

This book is for anyone with an interest in 3D printing and some basic computer skills. Whether you own a 3D printer or not, you can design for them. You will need Blender, a free 3D tool, and this book. With a little creativity, one day you'll hold in your hands something designed with a computer.

Conventions

In this book, you will find a number of text styles that distinguish between different kinds of information. Here are some examples of these styles and an explanation of their meaning.

Code words in text, database table names, folder names, filenames, file extensions, pathnames, dummy URLs, user input, and Twitter handles are shown as follows: Windows and unsure, just choose the MSI package option.

New terms and **important words** are shown in bold. Words that you see on the screen, for example, in menus or dialog boxes, appear in the text like this: Locate the **Blender** download button on the main page for the latest version of Blender and click on it.

> **NOTE**
> Warnings or important notes appear in a box like this.

> **TIP**
> Tips and tricks appear like this.

Reader feedback

Feedback from our readers is always welcome. Let us know what you think about this book— what you liked or disliked. Reader feedback is important for us as it helps us develop titles that you will really get the most out of.

To send us general feedback, simply e-mail <`feedback@packtpub.com`>, and mention the book's title in the subject of your message.

If there is a topic that you have expertise in and you are interested in either writing or contributing to a book, see our author guide at www.packtpub.com/authors.

Customer support

Now that you are the proud owner of a Packt book, we have a number of things to help you to get the most from your purchase.

Downloading the color images of this book

We also provide you with a PDF file that has color images of the screenshots/diagrams used in this book. The color images will help you better understand the changes in the output. You can download this file from https://www.packtpub.com/sites/default/files/downloads/3DPrintingDesignsFunandFunctionalProjects_ColorImages.pdf.

Errata

Although we have taken every care to ensure the accuracy of our content, mistakes do happen. If you find a mistake in one of our books—maybe a mistake in the text or the code—we would be grateful if you could report this to us. By doing so, you can save other readers from frustration and help us improve subsequent versions of this book. If you find any errata, please report them by visiting http://www.packtpub.com/submit-errata, selecting your book, clicking on the **Errata Submission Form** link, and entering the details of your errata. Once your errata are verified, your submission will be accepted and the errata will be uploaded to our website or added to any list of existing errata under the Errata section of that title.

To view the previously submitted errata, go to https://www.packtpub.com/books/content/support and enter the name of the book in the search field. The required information will appear under the **Errata** section.

Piracy

Piracy of copyrighted material on the Internet is an ongoing problem across all media. At Packt, we take the protection of our copyright and licenses very seriously. If you come across any illegal copies of our works in any form on the Internet, please provide us with the location address or website name immediately so that we can pursue a remedy.

Please contact us at <copyright@packtpub.com> with a link to the suspected pirated material.

We appreciate your help in protecting our authors and our ability to bring you valuable content.

Questions

If you have a problem with any aspect of this book, you can contact us at <questions@packtpub.com>, and we will do our best to address the problem.

Chapter 1. 3D Printing Basics

As cool as 3D printing is, there is a lot of hype around it, which sometimes causes confusion. Before starting to design for 3D printing, it's best to know a little bit about 3D printing technologies.

3D printing is a limitless technology in the sense that there is no end to the things it can make. Still, that doesn't mean that it can make anything without limitations. 3D printing can make things that no other manufacturing method can, but it has rules that need to be followed to ensure success. There are different types of 3D printing as well, and each type comes with its benefits and drawbacks:

In this chapter, we'll discuss:

- What is 3D printing?
- What types of 3D printing are there?
- How do FFF printers work?
- The anatomy of an FFF print.
- Supportless 3D printing and YHT.
- Wall thickness and tolerances.

What is 3D printing?

3D printing is cool. It seems as if not a day passes without another mention of 3D printing online in the news and media. Everyone is getting excited about 3D printing. But when you look deeper, it seems as if everything is being 3D printed, and anything could be. Does 3D printing

something make it better? What exactly is 3D printing?

In many ways, 3D printers are just tools, the same as any that you'd find in a wood shop or garage. These tools make cool things, but not on their own, and just because something is made with, say, an electric drill press, that doesn't automatically make it better than something that isn't. It's the things that people, like you, are doing with these tools that make them cool.

I'm not saying that 3D printing isn't cool by itself. 3D printing lets you create things, test them, change their design, and try something new quickly until you get it right. It makes things of incredible complexity and, because it's additive manufacturing, generates comparatively little waste. The availability of cheaper and faster 3D printers means that there's a chance that there's a 3D printer near you.

What defines 3D printing?

There are many different types of 3D printers, but what makes them all similar is that they build solid shapes from layers of materials, starting with an empty build area and filling it with the print. This is called **additive manufacturing**, and it produces less waste than other techniques, such as starting with a base material that is cut away to make the thing.

3D printers also benefit from being computer-controlled machines, also known as **computerized numerical control** (**CNC**) machines, meaning they do what they do with minimal human interaction after the design work is done. They can make many identical copies of a thing one right after the other, and the design

can be shared online so that others can make their own copies.

While all 3D printing shares come common features, there are several distinct types of 3D printing that vary in how they produce the print. **Fused filament fabrication** (**FFF**), **powder bed**, or **light polymerization**, for example, all accomplish 3D printing in very different ways, and each with their own strengths and weaknesses. What works in powder bed 3D printing might not work with FFF 3D printing, and the part you get from light polymerization might not be suitable for the same usage as those made with the other techniques.

What to design for?

It is the best practice to always design towards the strengths and weaknesses of the medium you'll be using. The projects in this series of books will focus on designing for FFF 3D printers, because they're inexpensive and more readily available than the others, and the parts made with FFF 3D printers are suitable for a wide variety of functional uses. Also, many of the techniques for FFF design transfer to the other types of 3D printing. But because FFF 3D printers have limitations, there will be some things you need to know first.

How do FFF printers work?

There tends to be a lot of variation within the family of FFF 3D printers. Some have their mechanisms exposed to the environment so that they're easy to repair, while some are protected with fancy covers so that they look good. Some have one extruder, while some have two or more. Some have fancy interface screens, and some require you to use a computer to access even the most basic functions. Yet, for all their variations, there are many similarities that all FFF printers share which define their type. Being familiar with how FFF 3D printers work will help you guide yourself while designing for them.

For FFF 3D printers, a computer takes a 3D model and translates it into commands that the printer can follow. The printer then takes a roll of plastic filament on a spool and uses a feeder mechanism to feed it into the hot end, where the plastic filament is melted and squirted out at a controlled rate onto the print bed, where the print is built up. The extruder head and print bed are moved relative to each other in 3 dimensions, using some sort of movement system in order to create the 3D model:
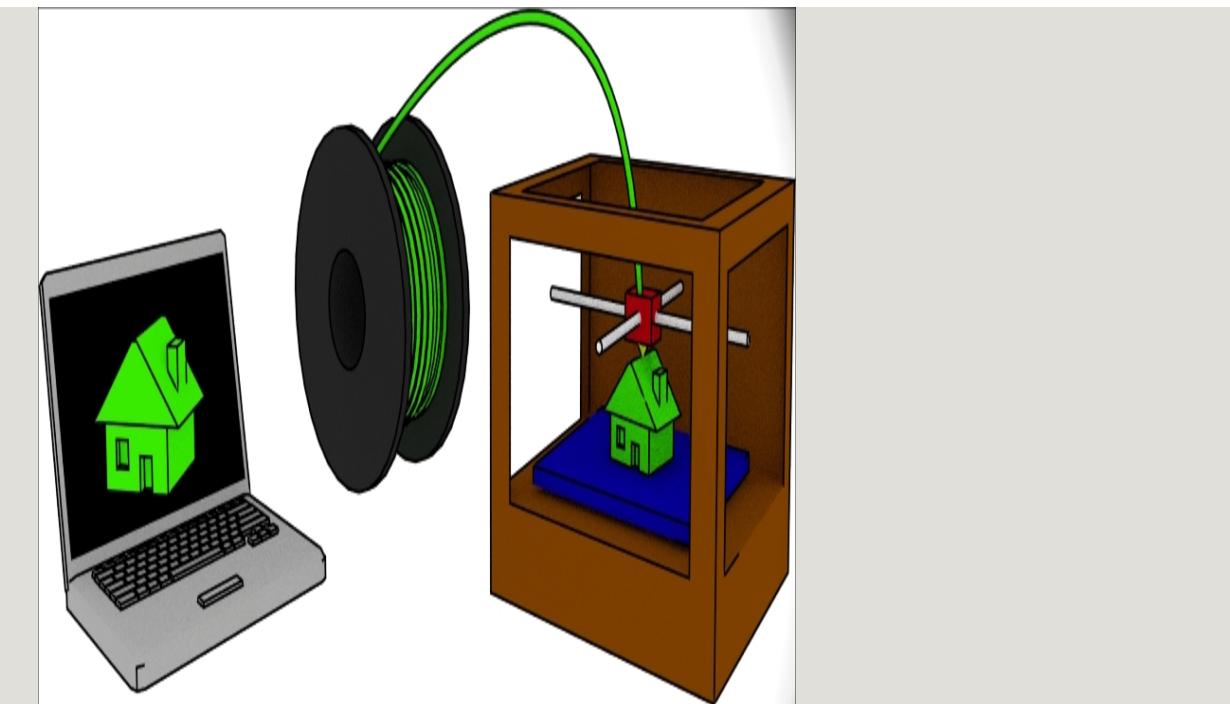
Drawing a print layer by layer like this takes, as might be expected, a little bit of time. The larger the object, the longer a print will take. FFF 3D printing isn't a fast process. But once the process is done, a new thing will have been created.

The anatomy of a print

Now that the mechanics of FFF 3D printing are clear, it's time to take a look at how a print is built. If an FFF print is stopped partway through, or observed during printing, the following can be seen:
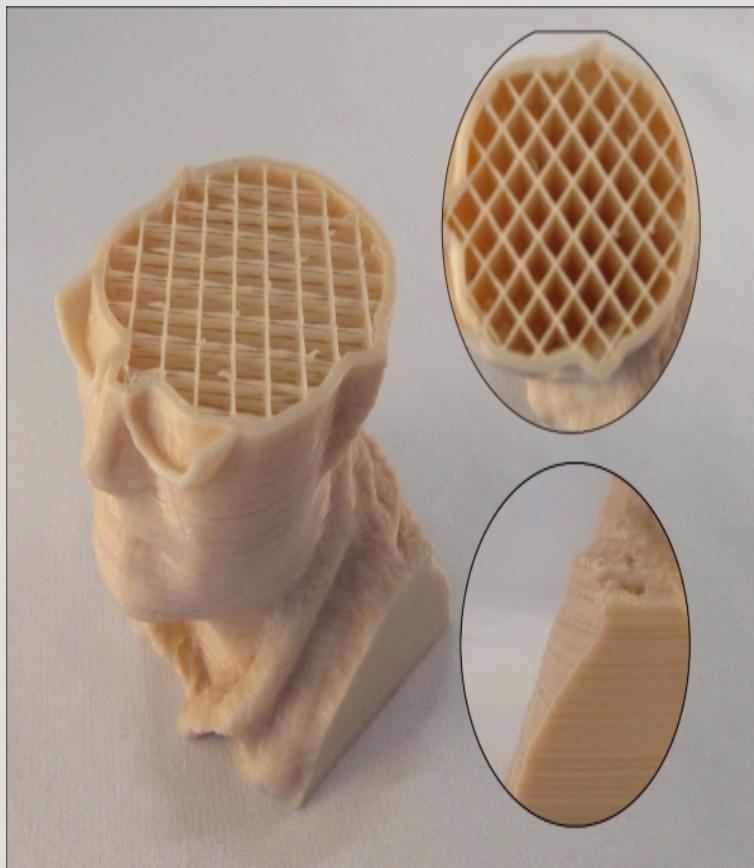
The following are the different parts shown in the preceding image:

- **Layers**: FFF prints in layers, with each layer sitting on the one below it. Prints can be made with thicker layers so that they print faster or thinner layers so that they look better.

- **Outlines**: When starting a layer, the outline of that layer will usually be printed first. FFF prints often have two or more outlines so that the outside of the print is strong.

- **Infill**: once the outline is done, the rest of the layer is filled in. If an area of the print will not be seen from the outside when the print is done, a loose infill is used to save material and give layers above something to sit on. Top layers are filled in completely. Most FFF prints are largely hollow.

FFF design considerations

The basic limitations of FFF printers stem from the fact that most FFF 3D printers are developed by people who have very little accountability. To the people creating and manufacturing these printers, if the printer can print a thing most of the time, then that's probably good enough. In this way, FFF printers are more like garage tools than desktop machines. For those unfamiliar with FFF printers, there are some drawbacks that need to be taken into account.

Overhangs and supports

FFF 3D printers have to worry about overhang. Overhang is when a part of the design, when it prints, will not have anything between it and the build platform. To compensate for this, the 3D printer can build a lattice of support material up to the overhanging part. After the print, the support material will have to be removed. But since for most FFF 3D printers the support material is made of the same material as the object, it can rarely be removed without a trace that is sometimes difficult to clean up completely and can leave a mess on more complex prints:

Because of the troubles with supports, it's a good idea to design for supportless 3D printing.

Supportless 3D printing

Think about building a snowman or sand castle. There's a lot that can be done with the medium of sand or snow, but try to get too fancy with the design and it will fall apart. As long as every part is sitting on top of something, chances are it will hold together. You could even slope gently outwards, as long as you don't push it too far.

It's the same with 3D prints. Because it prints in layers, each layer needs to have something to lay down on. If a design is made so that a part has nothing underneath it and is dangling in the air, then the printer will still extrude some plastic to try to print the part, but with nothing to print on, the plastic will just drool from the extruder until it gets wiped off on some other part, making an ugly mess and ruining the print.

As long as you put some thought into it, you can make designs that will succeed in most cases. There are a few rules that can help, and these rules can be illustrated with the letters Y, H, and T.

Y – gentle overhangs

Think about 3D printing a capital letter Y, standing up on the build platform—something like this:
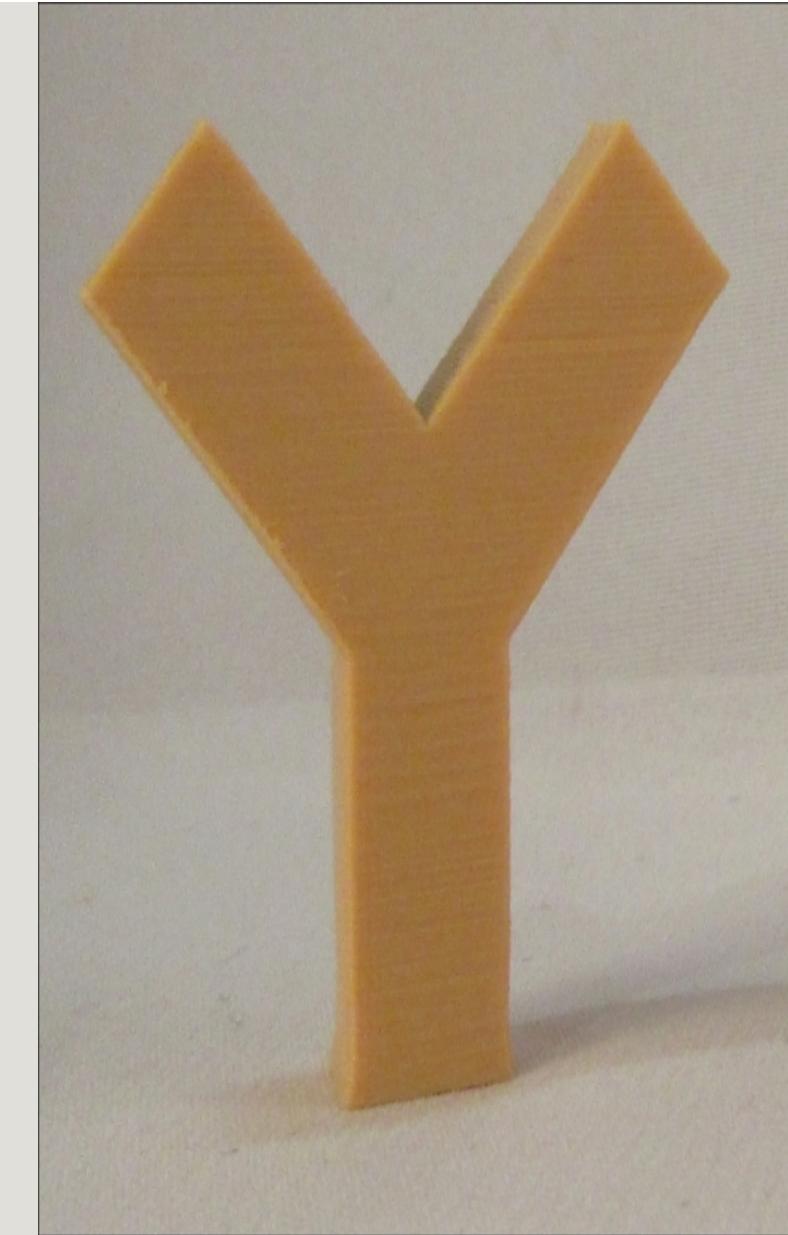
As the print gets to the part where the arms of the Y branch out, the change is gradual. It is possible to have the current layer slightly larger than the previous one, provided the overhang is gentle. Generally, a 45-degree overhang is safe. Hence, a shape like the letter Y will successfully print standing up.

However, if the overhang is too great or too abrupt, the new layer will droop, causing a print

to fail. Some 3D printer owners pride themselves in pushing their overhang and have seen success with angles as steep as 80 degrees, but to be safe, keep your angles no more than 45 degrees.

H – bridging

If a part of the print has nothing above it but has something supporting it on either side, like a capital letter H standing up, then it may be able to bridge the gap when printing:

Use caution when bridging. The printer makes no special effort when making bridges; they are drawn like any other layer: outline first, then infill. As long as the outline has something to attach to on both sides, it should be fine. But if that outline is too complex or contains parts that will print in midair, it may not succeed. Being aware of bridges in the design and keeping them simple is the key to successful bridging. Even

with a simple bridge, some 3D printers need a little bit more calibration to print it well.

Hence, a shape like the capital letter H will successfully print most of the time because of bridging.

T – orientation

If you were to try to print a capital letter T standing up on the build platform, you would surely run into problems:
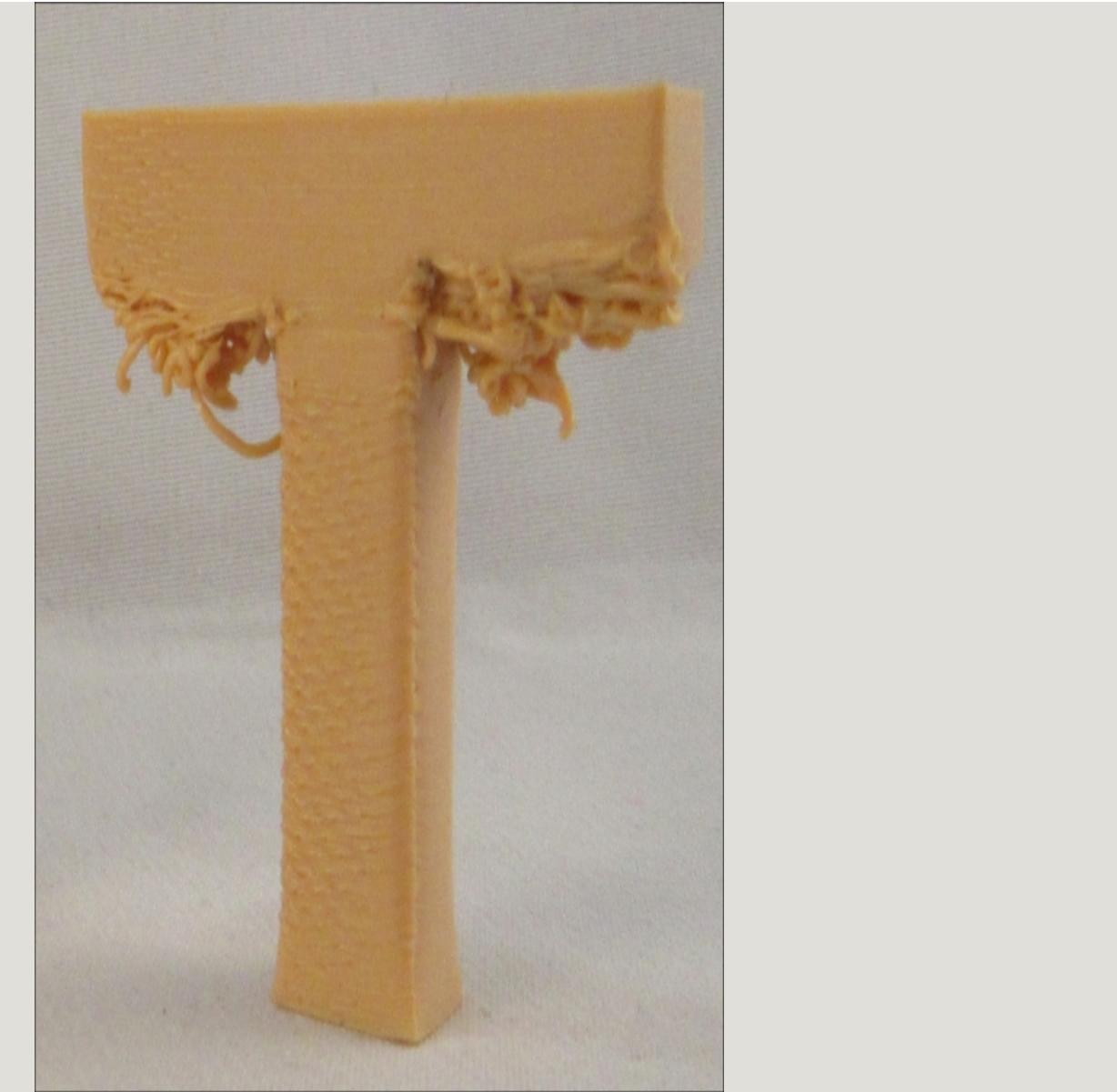
The top arms have far too much overhang to print successfully. Of course, the solution to this is simple: when designing, flip the T over or lay it down. In fact, every letter of the alphabet will print successfully if laid on its back, but the letter T illustrates this best. Sometimes, when designing a part for 3D printing, it's good to turn it around and orient it so that it prints well. Not every print needs to be printed in the same way it's going to be used.

Wall thickness

There is a minimum size of things that a 3D printer can print. This size is determined by the size of the hole in the nozzle, called the nozzle diameter. The most common nozzle diameter is 0.4 mm; however, most printers will not print a wall with a single extrusion thickness. They require that a wall be at least as thick as two nozzle widths, which in most cases means walls need to be at least 0.8 mm. However, because of the way slicers calculate outlines, 0.8 mm isn't just a minimum wall thickness—it's a target. For instance, if the wall is 1 mm thick, it won't be able to fill in the gap between the outlines, and there will be an air pocket. And while 0.4 mm is a very common nozzle diameter, it is not the only nozzle diameter, so a 0.8 mm wall may still be too thin for some 3D printers.

For thickness, it's best to err on the side of caution. A 2 mm wall is thick enough that slicers can use one or two outlines without conflict and still have room for a little infill, no matter the nozzle diameter. This will make solid prints that will succeed in almost all cases, and 2 mm is still fairly thin, allowing for considerable detail. Unless you are designing for a specific printer or planning to share your model with others, always make your walls a minimum of 2 mm thick to be safe.

Holes in models

Models for 3D printing must be closed, that is to say, they must have no holes in them. In a classic cartoon, there was a scene where bubbles were blown, but they were not bubble shaped. They were square, squiggly, and pink-elephant shaped. But no matter their shape, they were still

bubbles. If a hole developed in them, they popped. In the same way, models for 3D printing cannot have holes:
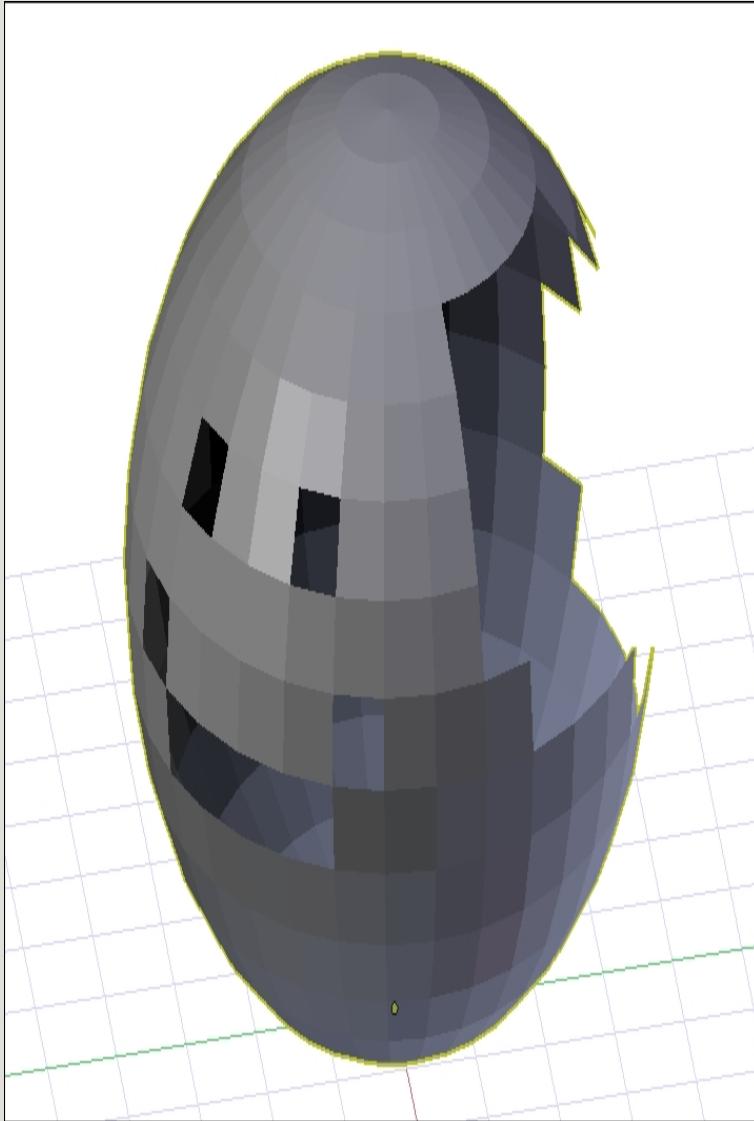
In mathematical terms, holes in models are included in a family of errors called non-manifold. Models for 3D printing must be manifold or else the slicer will have trouble telling what is supposed to be the inside and outside of the model.

In the same vein, a wall by itself, without an inside or outside, isn't printable because a 2D

wall has no thickness and doesn't describe a shape that can exist in real life. 3D prints must be part of a three-dimensional shape with a thickness, as described in the previous section.
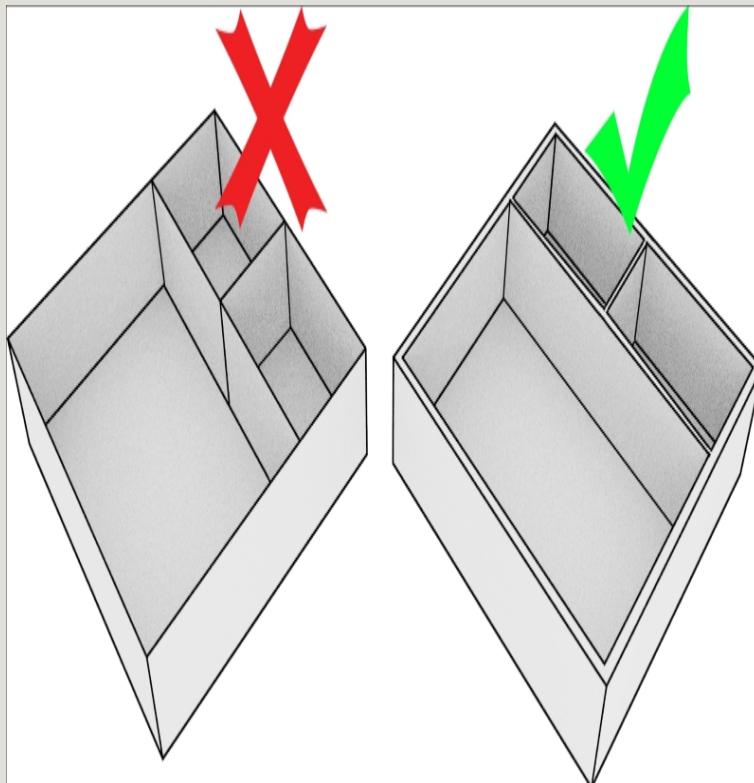
Summary

3D printing is cool and allows the creation of fantastic and detailed objects without needing much interaction with people after the design is done. But designing for 3D printing is a lot like designing for any other type of manufacturing. It helps to know a bit about the process involved and design with that process in mind.

Fused filament fabrication 3D printing, or FFF for short, is one of the oldest, most mature, and cheapest forms of 3D printing, so this series will focus on designing for it. It involves melting a plastic filament and drawing the object layer by layer, with each layer sitting on top of the one below it.

Designing for the most effective FFF printing means thinking about overhangs and supports and about the parts of the prints that don't have anything underneath them when they print. To avoid needing supports when printing, it can help to remember the letters Y, H, and T when designing, in order to remember to consider gradual overhangs, bridging, and orientation. In addition, it's important to remember that details should be, generally, about 2 mm thick.

Now that the mechanics of 3D printing and how they affect design have been covered, the next chapter will deal with the specific software that will be used in this series.

Chapter 2. Beginning Blender

3D printers need 3D models to print. Those models don't just come out of nowhere. You will need imagination, a little time, and software to create 3D models for the things you want to print. Never have there been more software options for creating 3D models—professional and free options. In this series, the software of choice is called Blender.

This chapter will introduce Blender, how to set it up, and some basic and mid-level functionality. Knowing the content of this chapter will get you over Blender's infamous learning curve and provide the basic knowledge and reference material necessary to follow along with future projects. We'll cover these topics:

- Why Blender?
- Downloading and installing Blender
- The default view
- The best settings
- Object creation
- Navigating the view
- Transforming
- Controlling transformations
- Selecting
- The Edit mode
- Blender to real life
- Exporting an STL

Why Blender?

With so many options for 3D modeling software, why would Blender, a software designed to make 3D animations, be the most popular choice?

The price is right

First things first: you don't have to pay for Blender. It is offered free of charge. If it works for you, you always have the option to donate, but Blender doesn't do anything to force this point. It is free now and forever.

Blender is comprehensive

While it's true that Blender is designed for animation because it covers everything from a blank canvas to a finished animation, it contains the ability to model objects, and it's one of the most robust suites of modeling tools anywhere. Learning Blender means that you may never need to learn another 3D modelling software.

It's getting better all the time

Blender is in constant development. If it doesn't have a feature you need, chances are that it may one day. Blender's developers are constantly responding to their audience.

But Blender isn't perfect

Despite Blender's advantages, it has a well-earned reputation for having a difficult-to-overcome learning curve. It's had a long and organic development cycle, which left it with a default user interface that isn't intuitive compared to most other software.

But Blender is also very configurable and, with a few simple settings, it can be made much easier

for the beginner to use. The rest of this chapter will help get you past Blender's short comings so that you can start developing awesome 3D models.

Downloading and installing Blender

The first thing that needs to be done is downloading and installing Bender. Follow these steps:

1. On your PC or Mac, open a your web browser and go to http://www.blender.org.
2. Locate the **Blender** download button on the main page for the latest version of Blende
3. r and click on it:

4. Scroll down and find **Download** under **Blender** and click
   on it. If you're on Windows and unsure, just choose the
   MSI package option.
5. When the installer finishes downloading, run it.
6. Follow the prompts to install **Blender**.
7. When the installer is finished, run **Blender**. Click
   anywhere to close the splash screen.

Blender is now installed and ready to use.

The default view

Blender's interface is made up of many smaller windows called **panels**. There are many different panels available in Blender. Like most things in Blender, the panels are completely configurable. Panels can be added or removed as needed, and panel layouts can be saved and switched among easily. For simplicity, the default view—the way Blender is presented the first time it loads up— will be used throughout this series. It provides most of the necessary functionality:
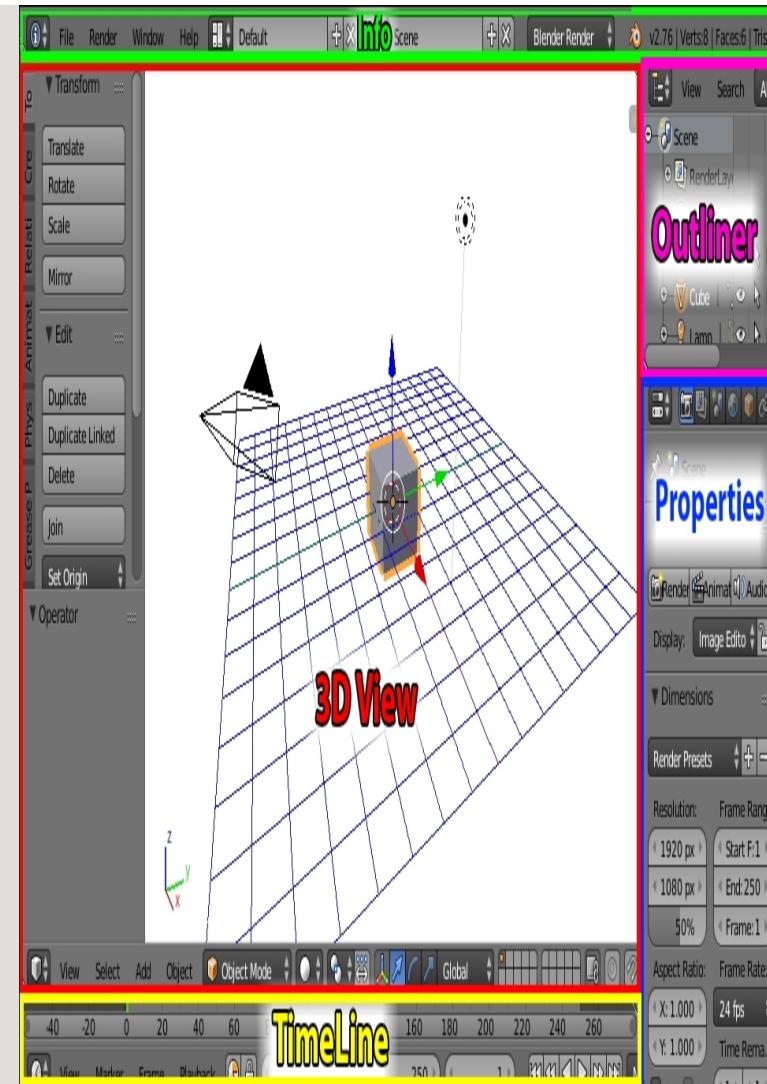
NOTE

For the most part, the screens shown in this book series will look similar to the default Blender screen. The major change will be to the background color of the 3D View, a choice made to make the illustrations more compatible with printing.

Here's an explanations of the different panels in the default view:

- **The Info panel**: Located across the top of the window, the **Info** panel has many of the menu options in most programs, such as **File**, **Window**, and **Help**. It also has **Layout** settings, **Scene** settings, and **Renderer** options, but we won't be using them much. Finally, there are specific details about the version of Blender, the current scene, and selected objects. If the **Info** panel is cut off,

bringing the mouse pointer over to the panel and using the scroll wheel will bring the information into view.

- **The Outliner**: Located in the upper-right section of the window, the **Outliner** contains the list of all objects.

- **The Properties panel**: Located in the lower-right section of the window, the **Properties** panel is broken into many tabs related to the currently selected object. Which tabs and properties are available will change depending on what is selected. If the tabs are cut off, bringing the mouse pointer over them and using the scroll wheel will bring them back into view.

- **The Timeline**: This panel is largely unnecessary for the purposes of this book series, so it can be removed or ignored.

- **The 3D View**: Taking up most of the screen, the **3D View** is where most things will be happening, and it consists of many smaller parts itself.

Each of the panels has its own keyboard shortcuts. In order to use them, the mouse pointer must be over the panel. If a keyboard shortcut isn't working, it's probably because the mouse pointer isn't in the right place.

The 3D View

Because it's so complex and important, the **3D View** will be given some special attention.

The **3D View** is where most of the action takes place, and it has the most visual feedback of the work. The **3D View** has its own menus and panels:
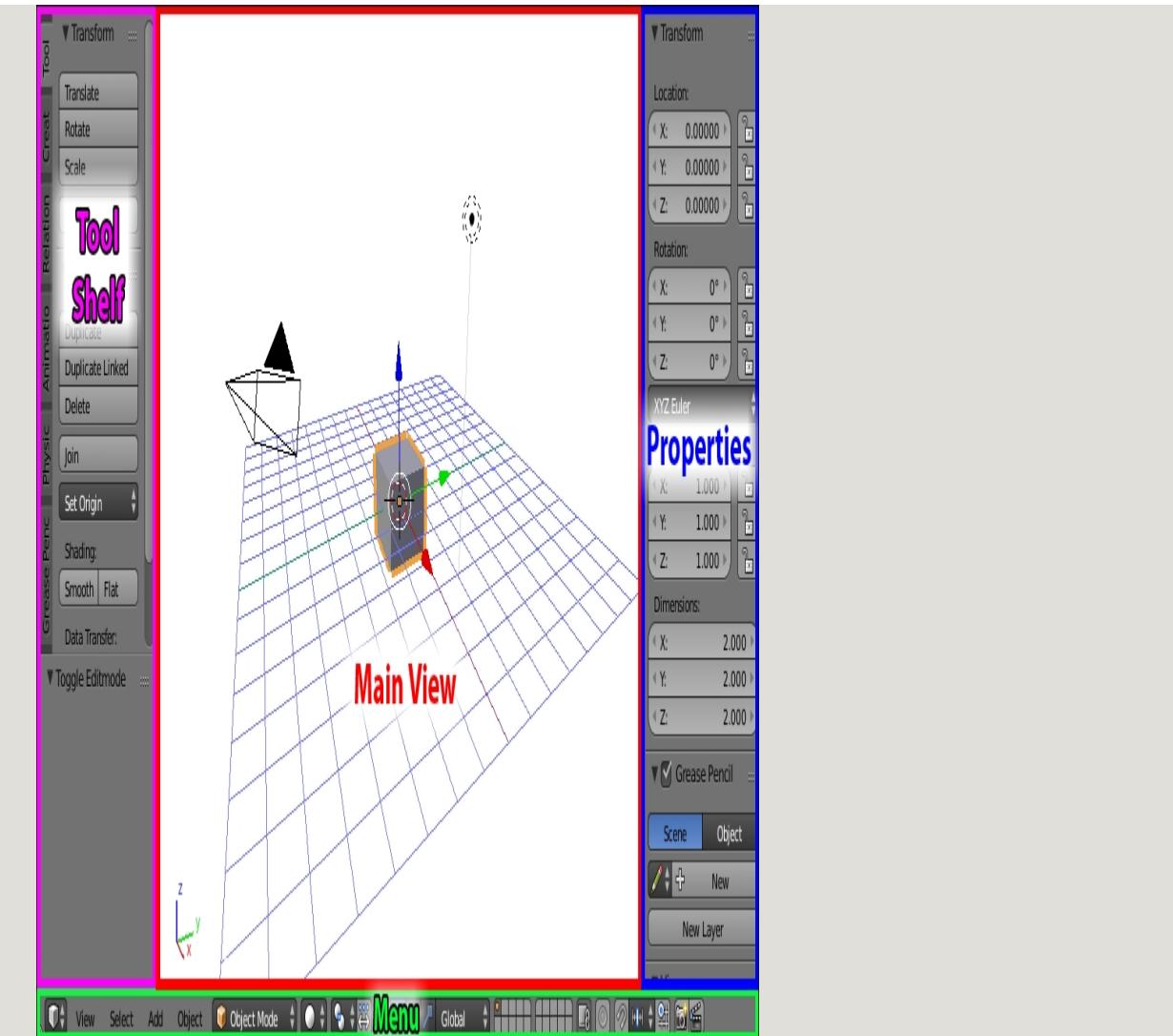
Here's what the panels do:

- **Menu**: At the bottom of the **3D View** is its **Menu**. It contains options and commands specific to the **3D View**. If the menu is cut off, bringing the mouse pointer over the panel and using the scroll wheel will bring the cut-off information into view.

- **Tool Shelf**: On the left-hand side is the **Tool Shelf**. It contains a tabulated set of buttons that can be used to do many useful things and change depending on the object selected. The bottom half of the **Tool Shelf** contains operator options. Any time an operation is performed, this area will be populated with options for that operation that can be edited until the next operation is performed. The **Tool Shelf** can be hidden or revealed by using the **3D View** menu and selecting **View | Tool Shelf** or by pressing *T* key on the keyboard.

- **Properties**: On the right-hand side is the **Properties** panel. Hidden by default, this panel contains information about the currently selected object as well as other options about the scene in general. The panel can be hidden or revealed by using the **3D View** menu and selecting **View | Properties** or by pressing *N* key on the keyboard.
- **Main View**: Of course, in the middle is the view of the current scene with all the visible objects.

The 3D cursor

In the **3D View**, there is a little red-and-white circle that starts in the middle of the 3D space. New objects will be created wherever the 3D cursor is located, and it's very easy to move the 3D cursor accidently, so it'd be good to know how to put the 3D cursor back in the middle and how to move it on purpose. The 3D cursor can be manipulated by:

1. Clicking the right mouse button in the **3D View** to move the 3D cursor wherever it is clicked in a plane relative to the view, which can be unpredictable.
2. Navigating to **View | Align View | Center Cursor and View All** or pressing *Shift + C* on the keyboard while in the **3D View** to put the 3D cursor back in the middle quickly. This is the easiest way to fix the 3D cursor if it gets moved unexpectedly.
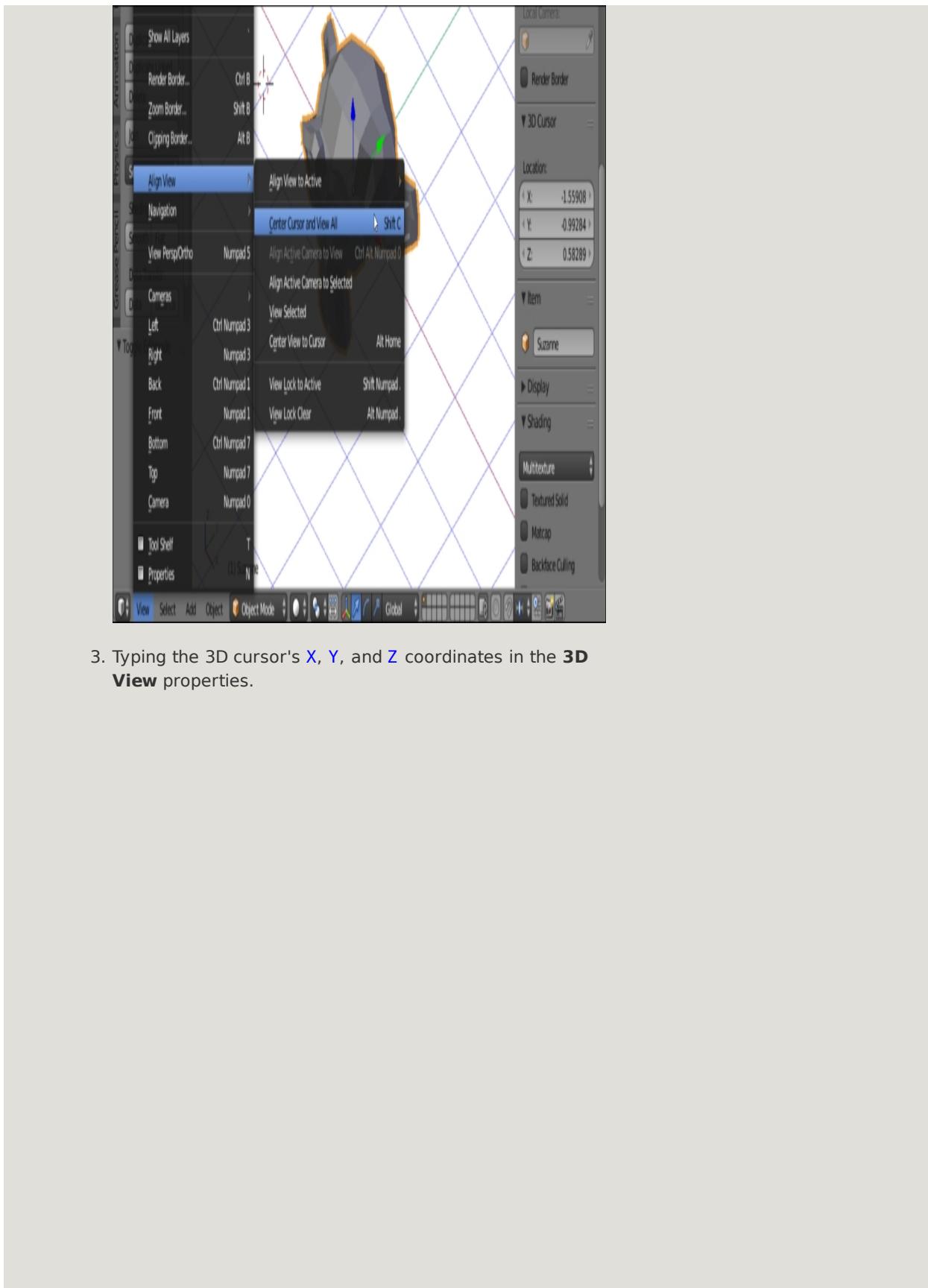
3. Typing the 3D cursor's X, Y, and Z coordinates in the **3D View** properties.

The best settings

Blender has a reputation for being difficult and unintuitive to use. This is true for the default settings. However, Blender is also very easily adjustable with a few settings and can be made much easier to use. How you set up your instance of Blender will depend on what your setup is like.

To access the settings, in the top menu select **File** | **User Preferences**. In the Blender **User Preferences** panel that comes up, select the **Input** tab button:
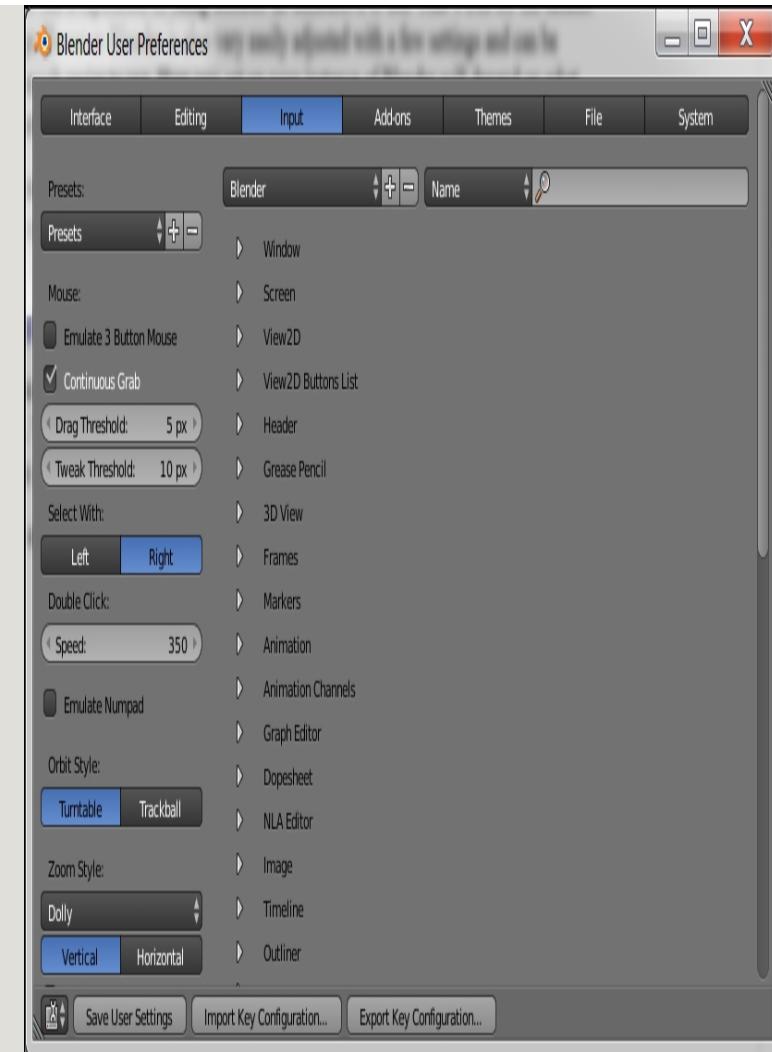
A scroll-wheel mouse and number pad

The recommended setup for Blender is to use your keyboard and a mouse with a scroll wheel. In this case, there is only one setting that is recommended to be changed from the default:

1. Click on the button that says **Left** under the words **Select With**.
2. Click on the button that says **Save User Settings**:
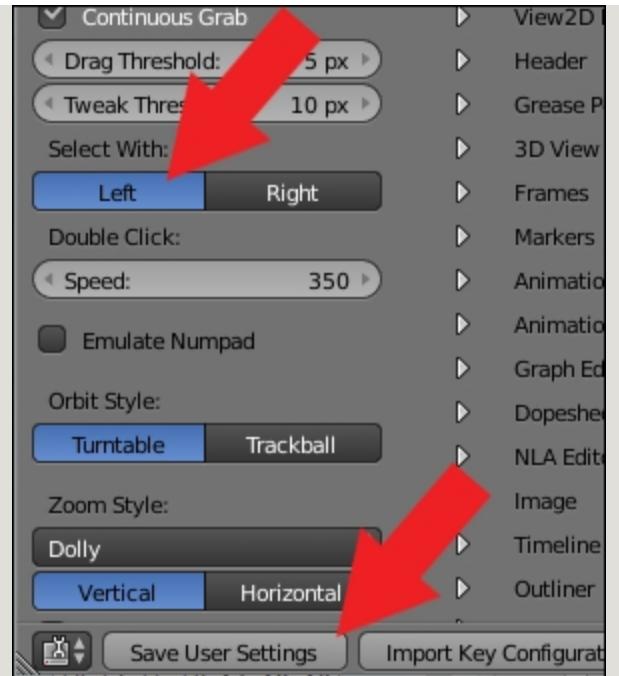
Changing this one setting will make Blender much more intuitive to use.

Because the default is for the right mouse button to be the select button, and some may prefer to keep this default, this book will refer to whatever option is chosen as the select mouse button, and the other button as the not-select mouse button. While still a bit confusing, it will help users who miss this section.

With a scroll-wheel mouse, the scroll wheel can be depressed for a middle mouse button click. Blender uses this middle mouse button to manipulate the view.

If you have a Numpad but no scroll-wheel mouse, it is still recommended to use these settings because the interface will be much more intuitive, although you will be sacrificing some functionality that the middle mouse click offers. The choice is yours whether it's worth having a more intuitive interface.

A laptop with a touch pad and no number pad

On a laptop with a touchpad (with no middle click) and no number pad, both very important for navigating the view, select the following settings:

- **Emulate 3 Button Mouse**
- **Emulate Numpad**
- Click on the button that says **Save User Settings**
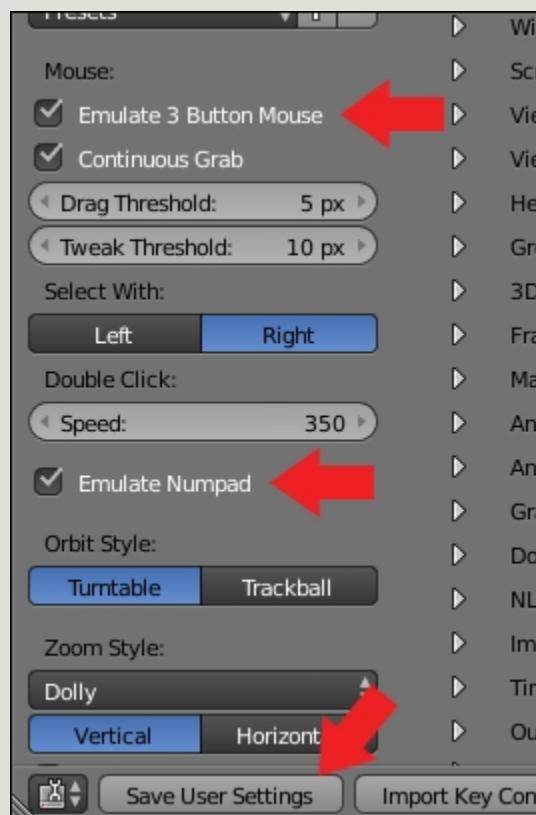
With this setup, most of Blender's functionality is available to laptop users, although these settings are less intuitive. With these settings, you will need to use the right mouse button to select objects, *Ctrl* + right mouse button as a middle mouse button to change the view, and the number keys across the top will perform the function of the number pad on a regular keyboard.

With Blender set up, it's time to start learning to use it. Close the **User Preferences** window.

> **NOTE**
>
> Blender users tend to use keyboard shortcuts for almost everything. It's recommended that you use Blender with one hand on the mouse and the other on the keyboard. Learning and using keyboard shortcuts will speed up your process and, with just a little practice, will become second nature. Going forward, all methods of accessing commands will be taught. Try to practice using the keyboard shortcuts.

Object creation

Most tutorials for Blender start with navigating the **3D View**. But in Blender, the default scene is kind of boring for this, just a cube that looks the same from every angle. Instead, let's make something more interesting to look at.

First, let's look at how to clear the default scene:

1. Select all the objects in the scene by going to the **3D View** menu and choosing **Select** | **(De)select** all twice or pressing the *A* key twice. Everything in the scene should have an orange line around it. If not, do it again.
2. Delete everything by going to the **3D View** menu and choosing **Object** | **Delete** or pressing the *X* key.

Your **3D View** should now have nothing but the grid, which can be thought of as the floor of the scene.

To create an object, go to **3D View** menu, choose **Add** or press *Shift + A*.

In the menu that pops up are all the basic objects that can be inserted into the scene. Blender offers many basic shapes that can serve as a starting point for anything you want to create. Sometimes, a basic object is all you need, and sometimes, the basic object needs to be modified.

In this case, add a **Monkey** object to the scene:

1. In the **3D View** menu, choose **Add** or press *Shift + A*.
2. Select **Mesh** | **Monkey** in the menu that appears.
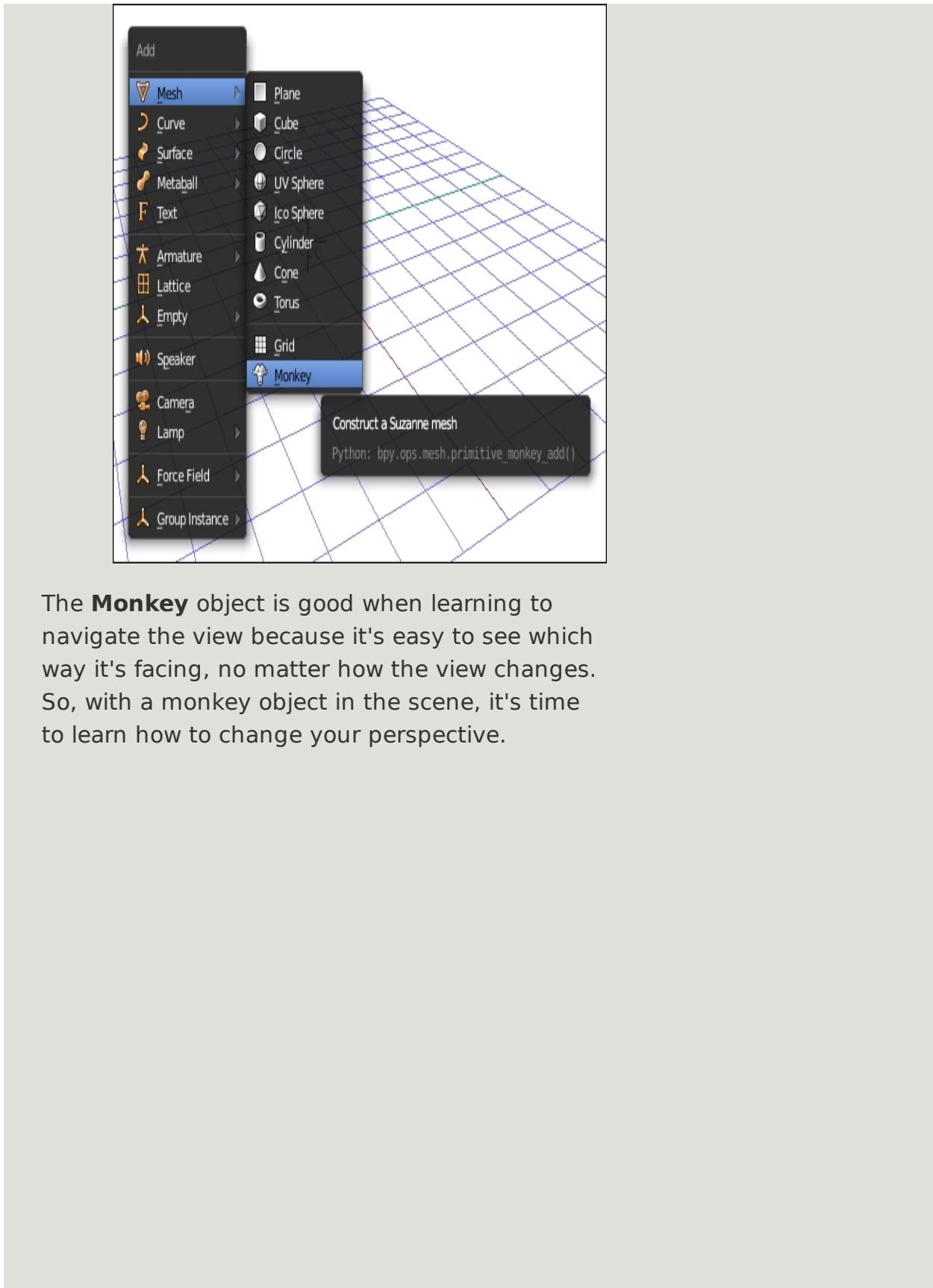
The **Monkey** object is good when learning to navigate the view because it's easy to see which way it's facing, no matter how the view changes. So, with a monkey object in the scene, it's time to learn how to change your perspective.

Navigating the view

Because Blender is all about working in 3D, but computer screens are flat, it is important to know how to change how you're looking at something in Blender.

> **NOTE**
>
> All of the following commands can also be found in the **3D View** menu under **View** | **View Navigation**; however, since adjusting the view happens so frequently, it is recommended to learn the mouse and keyboard shortcuts instead of navigating menus to do this.

Rotating the view

In Blender, you can change the angle of the view by:

- Clicking and holding the middle mouse button and moving the mouse pointer.
- Pressing *2* or *8* on the number pad to rotate the view up and down.
- Pressing *4* or *6* on the number pad to rotate the view left and right.

The point at which the view is rotating can change. Follow these steps to center the view rotation on a specific object:

1. Select the object.
2. Press the . (period) key on the number pad.

The selected object will fill the view and all view rotations will now center on that object:

User Persp

(1) Suzanne
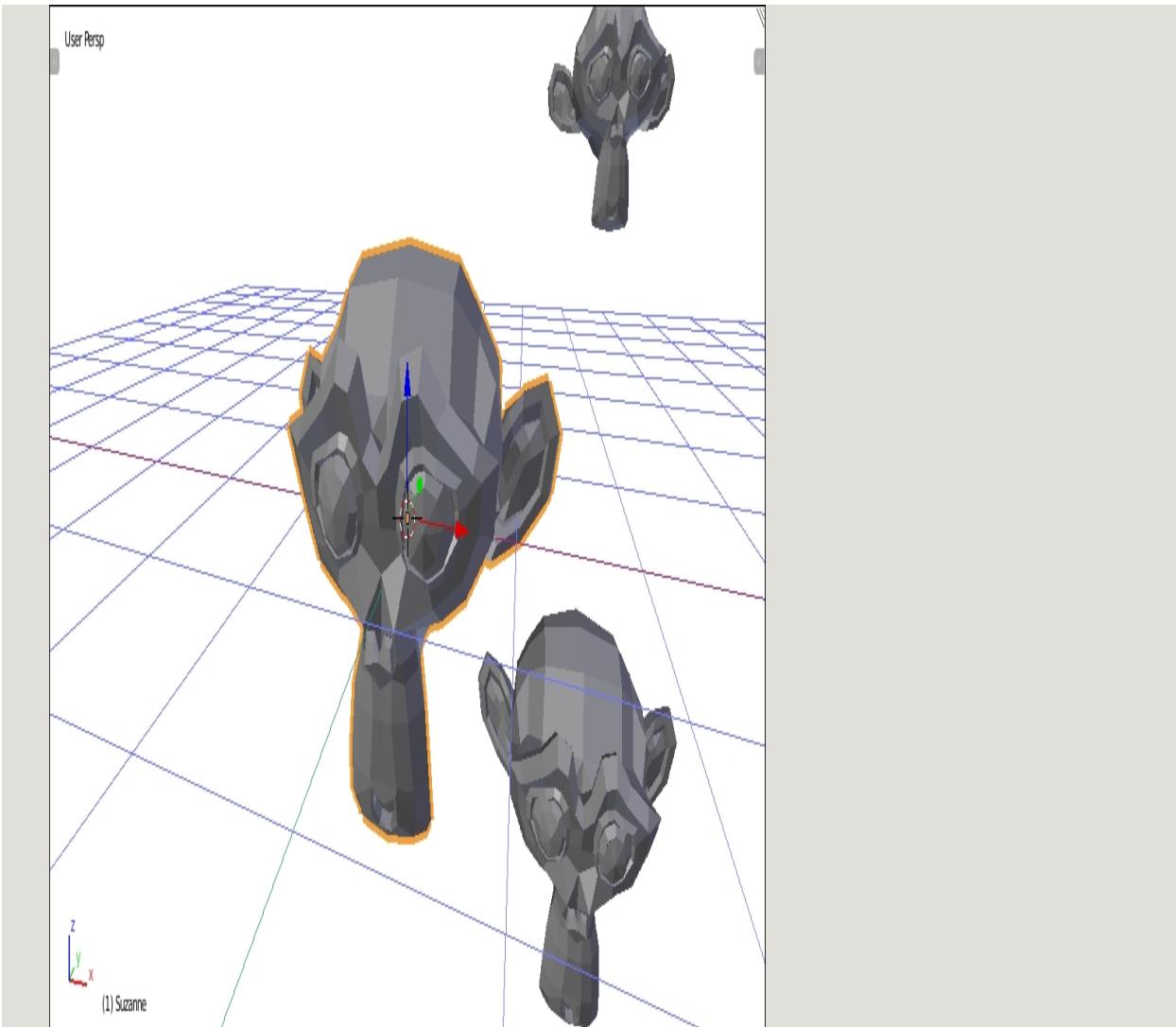
Jumping to rotation

Blender has six set views that can be quickly jumped to at any time:

- Press *1* on the number pad to jump to the front view
- Press *3* on the number pad to jump to the right view
- Press *7* on the number pad to jump to the top view
- Press *Ctrl + 7* or *9* on the number pad to jump to the bottom view
- Press *Ctrl + 1* on the number pad to jump to the back view
- Press *Ctrl + 3* to jump to the left view
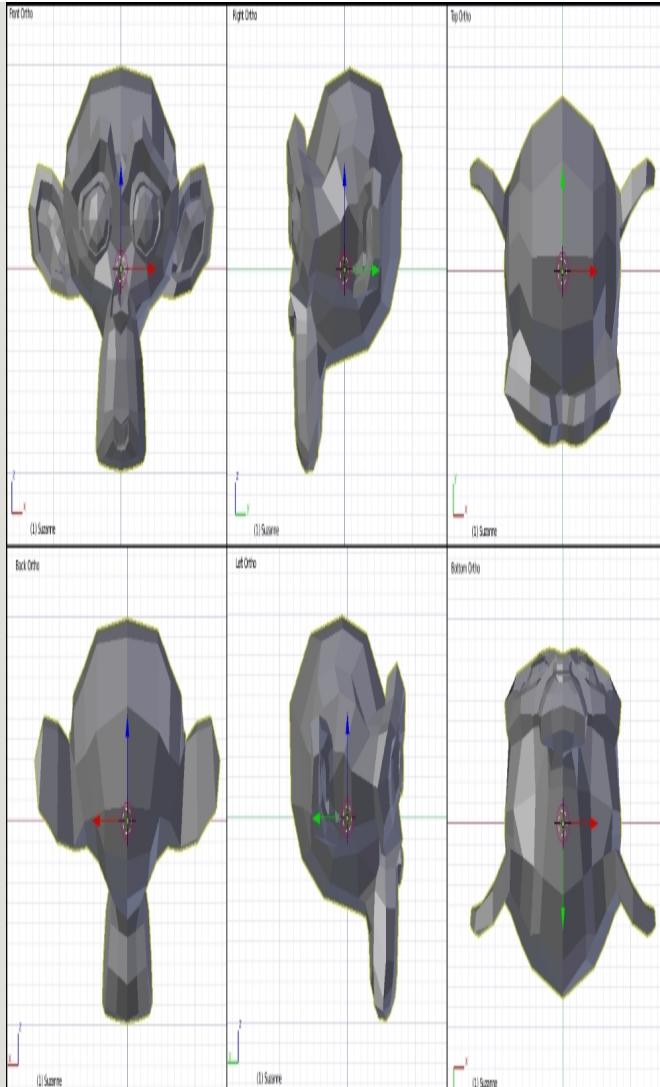
Panning the view

Panning the view means moving without changing the rotation, like moving your head while it's still pointed in the same direction. To pan the view, follow these steps:

- Press *Ctrl + 4* on the number pad to pan left
- Press *Ctrl + 6* on the number pad to pan right
- Press *Ctrl + 8* on the number pad to pan up
- Press *Ctrl + 2* on the number pad to pan down
- Hold *Shift* while clicking and holding the middle mouse button and move the mouse

Zooming the view

Zooming the view is moving closer to or farther from the object. To zoom the view, do one of the following:

1. Turn the scroll wheel.
2. Press Ctrl + middle mouse button and move the mouse.
3. Press *+* or *–* on the number pad.

Orthographic versus perspective view

To toggle between orthographic and perspective views, press *5* on the number pad.

The words **Persp** or **Ortho** can be seen in the upper right-hand corner of the **3D View**, indicating which view is being used:
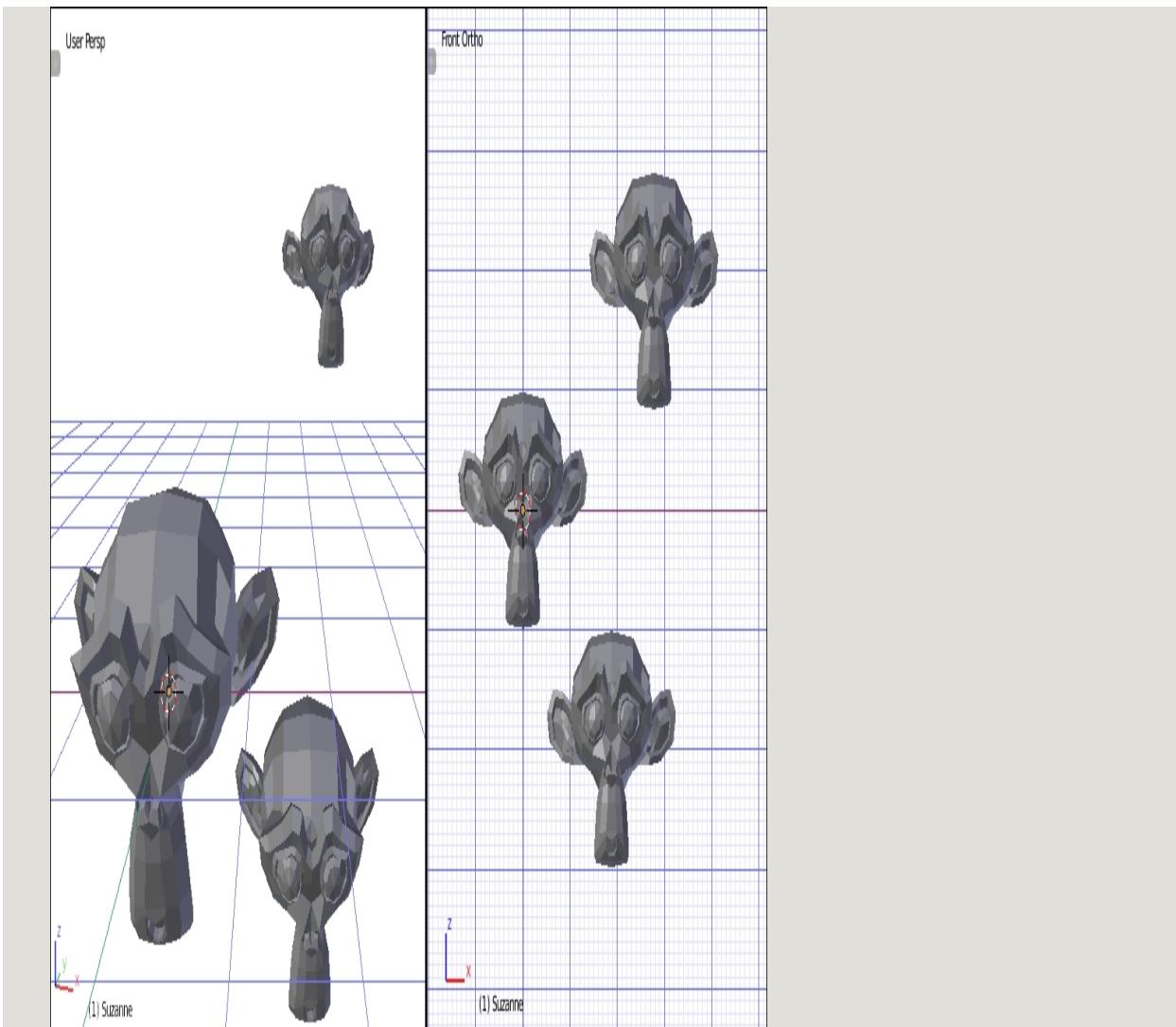
There are two ways of looking at the **3D View** on a computer. Perspective is the default, and more closely resembles how things will look when viewed through a camera or our eyes: closer objects look bigger; farther objects look smaller. Orthographic makes everything the same size, no matter how close or far it is. This makes it easier to compare objects and determine their location relative to one another.

Because a lot of 3D modelling requires precision, orthographic is generally preferred; however, it can be confusing when two objects overlap exactly. For this reason, it's good to adjust the

view frequently and be sure that the action you're preforming is the action you think you're performing.

Wireframe and solid view

With 3D modelling tools, it helps to be Superman. Being able to see through objects can help when selecting and modelling things. You can switch between the **Wireframe** and **Solid** views using one of these methods:

1. Press *Z* to toggle between **Solid** and **Wireframe** view.
2. On the **3D View** menu, click on the **Method to Display** popup and choose **Solid** or **Wireframe**.
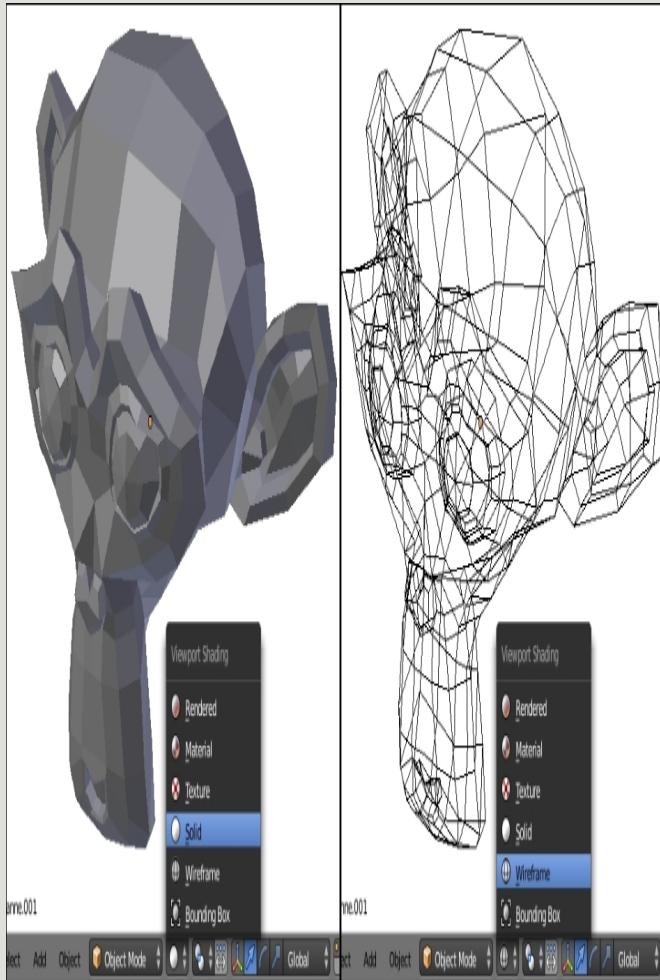
Wireframe mode is very powerful, especially in **Edit** mode, but it can be very confusing,

especially as objects get more complex. **Wireframe** mode allows objects behind the objects in the foreground to be selected without adjusting the view. Get used to switching between **Solid** and **Wireframe** mode frequently.

Transforming the object

In Blender, transforming an object changes the size, direction, or location of a thing without changing its shape. There are three basic transformation commands that are used frequently: Grab and Move, Rotate, and Scale.

To transform an object, be sure the object is selected, and then, from the **3D View** menu:

- Select **Object** | **Transform** | **Grab/Move** or press *G* on the keyboard to move the object
- Select **Object** | **Transform** | **Scale** or press *S* on the keyboard to scale the object
- Select **Object** | **Transform** | **Rotate** or press *R* on the keyboard to rotate the object

Then, move the mouse or use the arrow keys to transform the object. When the transformation has been accomplished, press *Enter* or the select mouse button to end the operation, or the transformation can be cancelled by pressing the not-select mouse button or the *Esc* key. The following diagram shows these object transformations:
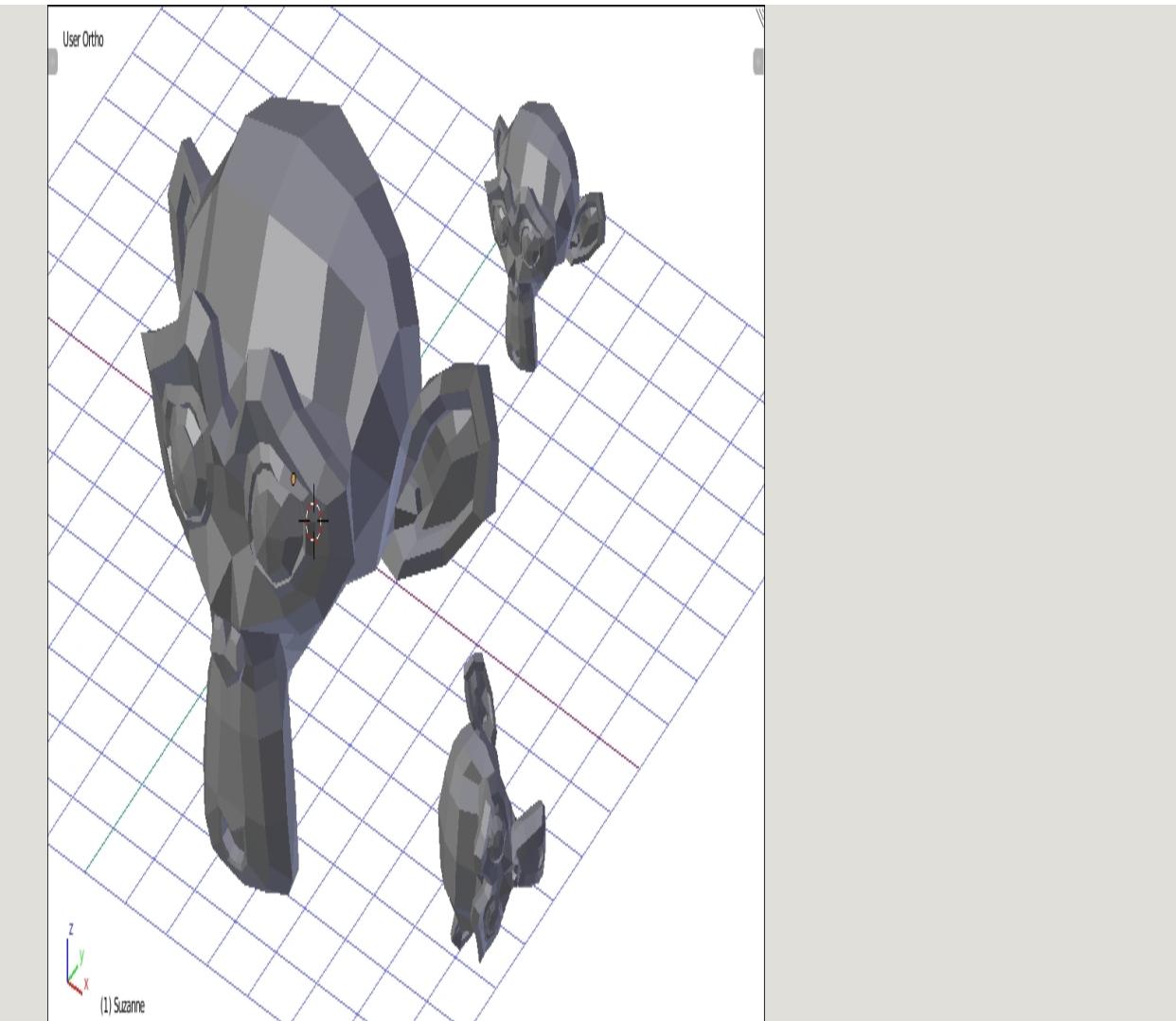
Transformations can also be undone after they're completed by pressing *Ctrl + Z.*

NOTE

Transform the **Monkey** added to the scene by moving, rotating, and scaling it. Get a feel for how these commands work. Then, try to move the **Monkey** to a specific place in the scene. When you think you have it, change your view to see whether it's really where you think it is.

Controlling transformations

By default, operators in Blender operate on a 2D plane tangential to the view. This is a fancy way of saying that without any additional controls, it can be hard to predict how a movement or rotation operation will work. For instance, moving something in a random view can include moving up and down more than expected. This effect won't be clear until the view is changed and the transformation is inspected from a different angle, as shown here:
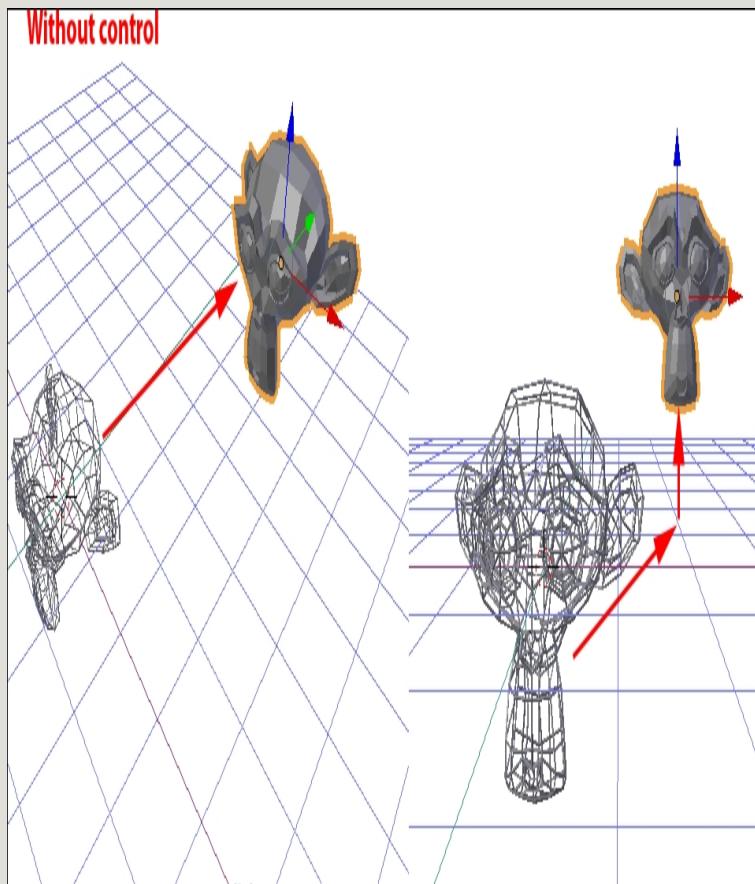
If you can't predict how operations will work, it can be hard to make the things you want. So it is

very important to be able to control transformations.

There are two main ways of controlling operators: **controlling the view** and **axis locking**.

Controlling the view

The first way to control the operation is by controlling the view. By default, operators depend on the view, so by controlling the view, you can control the action. For instance, an object, when added to a scene, is exactly halfway through the grid plane of the world by default. If you want to make sure the object stays halfway through the grid plane when moving, follow these steps:

1. Select a newly added object.
2. Switch to **Top Ortho** view by pressing *7* on the number pad.
3. Grab and move (*G*) the object around.
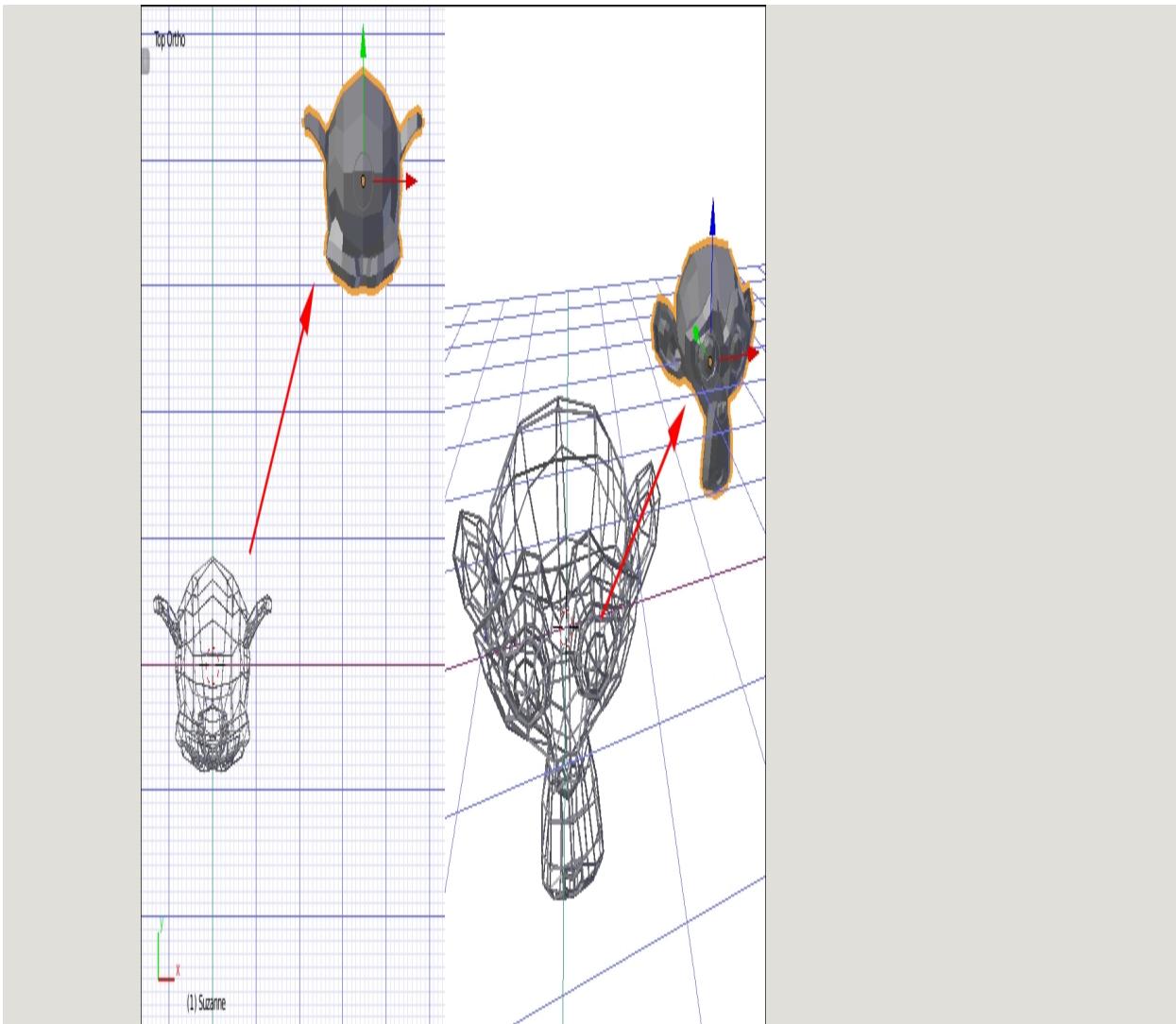
Adjust the view to see how exactly how the object moved. When moved while in the top view, the object stays on the grid plane. This is because from the top view, only forward, backward, and side-to-side motion is possible.

In the same way, moving objects while in the side views will only move them forward and back and up and down, and while in the front or back view, they will only move side to side and up and down.

Likewise, rotating depends on the view. When viewing from the top, rotation will be like, a top

spinning around its middle. If viewed from the side when doing rotation, it will flip over its middle, and if viewed from the front, it will roll side to side around its middle.

Controlling the view is a quick and easy way of controlling movement and rotation, but always be sure to adjust the view to make sure things are happening the way you expect. However, to control scaling, as well as to control movement and rotation independent of the view, there is another way.

Axis locking

Blender has special commands for changing the behavior of operations. These commands are available while performing transformations.

The first thing to understand is about the three axes that we'll be locking along. 3D refers to the three dimensions, or unique directions, that things happen in. The dimensions are called by the letters X, Y, and Z. There is an illustration in the corner of the 3D View that shows which way is which, and is always pointing in the right direction:
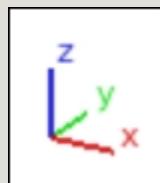
In Blender, and generally in 3D printing, X is side-to-side motion, Y is back-and-forth motion, and Z is up and down. In the previous screenshot, the X axis is red, Y is green, and Z is blue.

When performing a transformation, you can lock the transformation to the axis you want by starting the transformation as mentioned previously, and while moving the mouse or using the keyboard, doing the following:

- Pressing *X*, *Y*, or *Z* on your keyboard to lock the transformation to the desired axis

- Pressing *Ctrl + X*, *Ctrl + Y*, or *Ctrl + Z* to lock the transformation to all but the chosen axis

- Holding the middle mouse button and moving the mouse to choose an axis to lock the transformation to
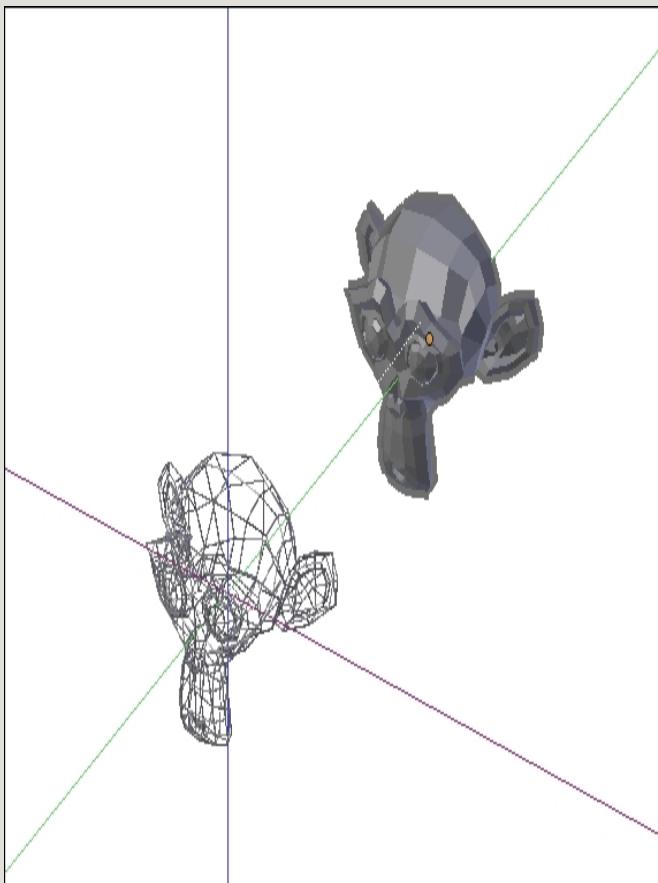
NOTE

Try out axis locking with movement, rotation, and scaling, and adjust the view to see what effect it has. Notice that with axis locking, the scale function has some additional abilities that aren't possible just by controlling the view. It actually only scales in the chosen axis. This can be very powerful while adjusting the shape of objects.

Precise transformation

Finally, under the category of controlling transformations, during transformation operations, the transformation can be precisely controlled by typing in a number that will relate to the operation being performed or edited afterwards in the operation properties in the tool box, on the left side of the **3D View**. In other words, to move something exactly two units up, for example, follow these steps:

1. Select the object.
2. Start the movement transformation (*G*).
3. Type *Z* to lock the movement to the Z axis.
4. Type *2* on the keyboard.
5. Press *Enter* or click the select mouse button to finish the operation.
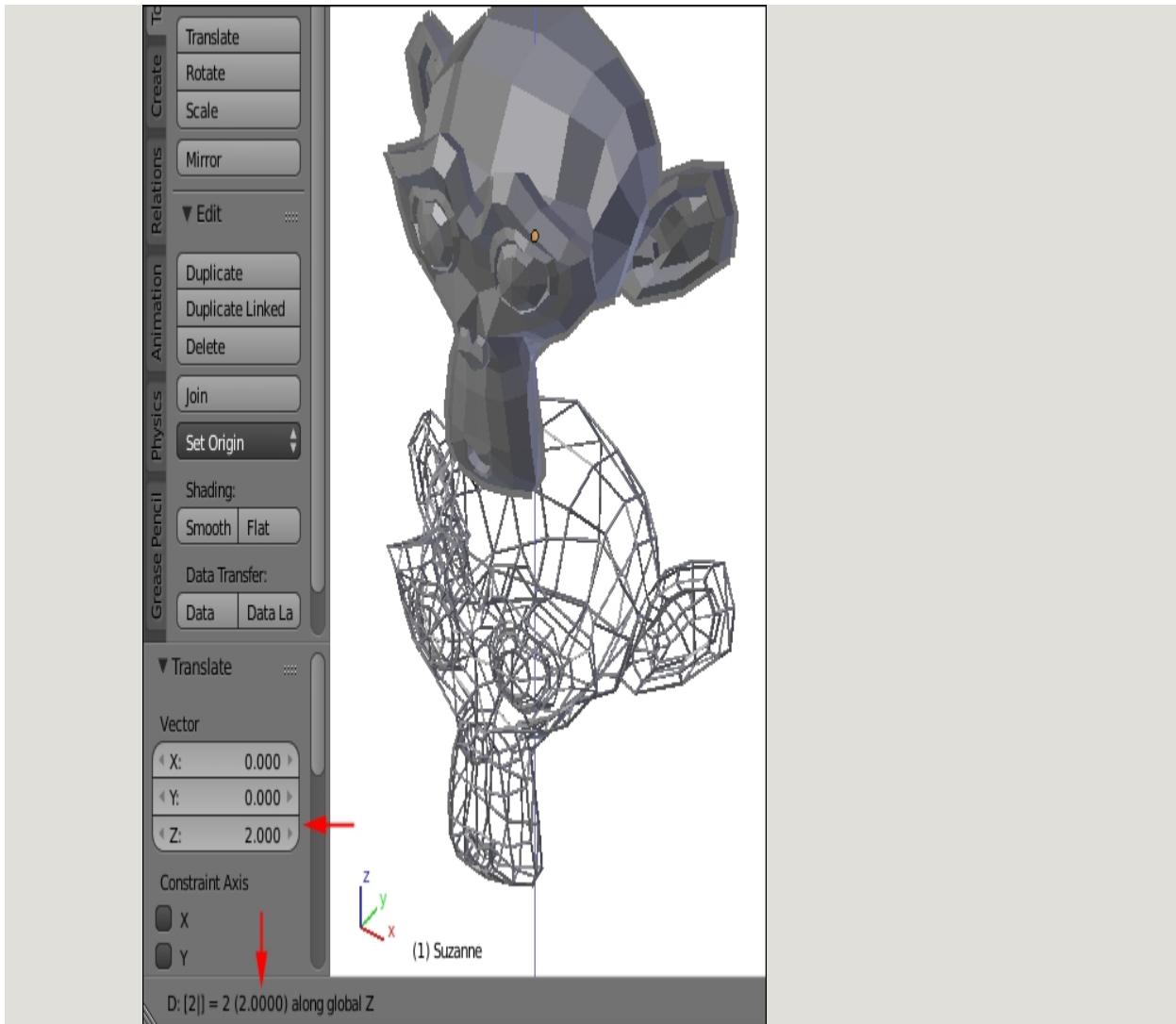
Typing commands has different effects for different commands, as follows:

- When moving, typed commands are used to state the number of units an object will move along the selected axis. For example, *2* along the Z moves it two units up, and -2 along the Z moves it two units down.
- When scaling, typed commands specify the scale factor. 1 means no change, 2 means twice as big, and 0.5 means half size.
- When rotating, typed commands specify clockwise degrees. For instance, 180 turns it around backwards.

Typed commands can include negative numbers and decimals and can be edited with the *Backspace* key. While performing a

transformation, the menu for the **3D View** changes to a description of the transformation, including the typed units, which can be useful when editing your typed commands.

Origin manipulation

Objects in Blender have an origin. Origins are depicted as a dot and start out in the middle of the object. Individual object transformation commands take place relative to the object's origin. It's possible that the origin can be moved as a result of editing, which can cause unexpected results when rotating or scaling the object. The origin can also be moved on purpose to control the effect of modifiers.

The origin can be reset using the origin controls. The origin controls can be found by one of the following methods:

1. In the **3D View** menu, choose **Object** | **Transform** and find the origin controls.
2. In the **Tool Shelf**, find the **Set Origin** dropdown and choose the desired option.
3. On the keyboard, press *Ctrl + Shift + Alt + C.*
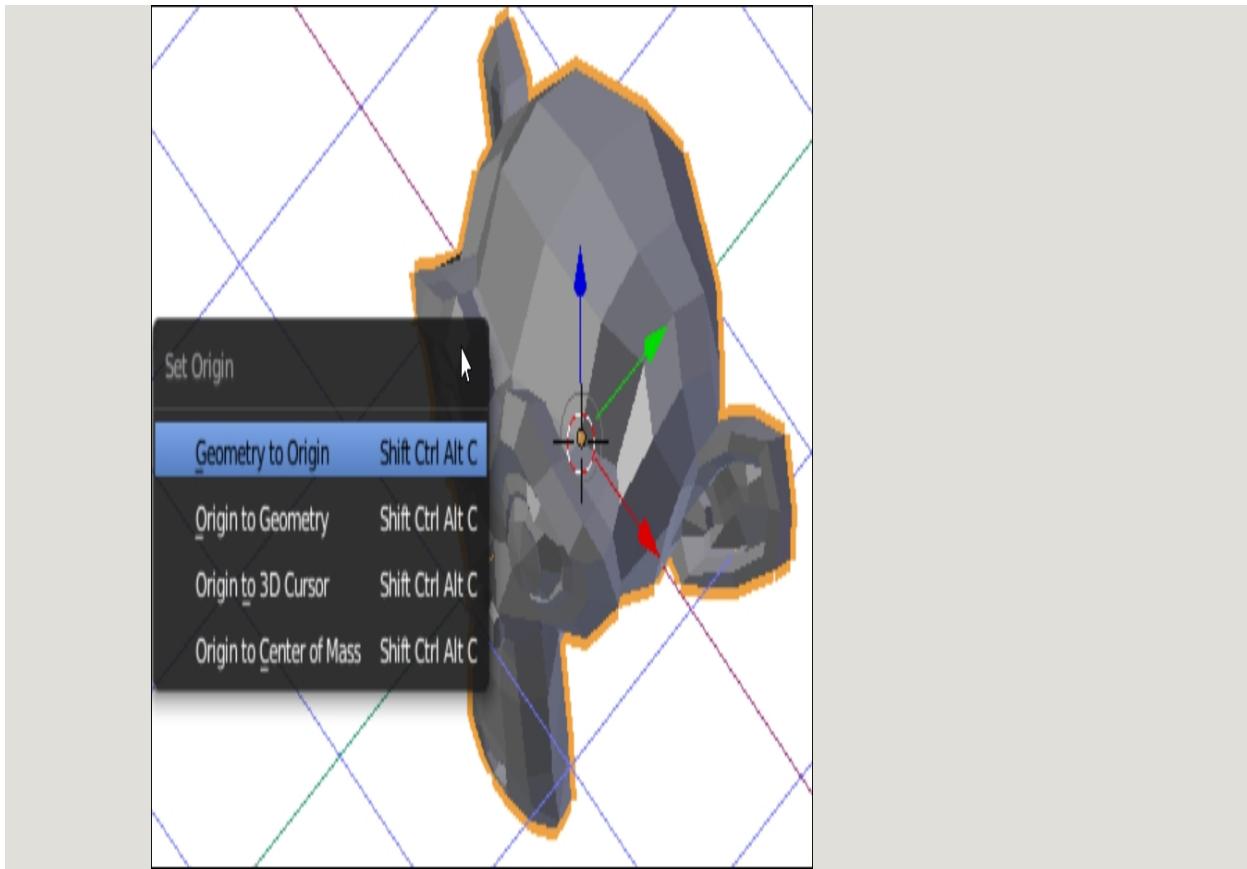
These are the commands to manipulate the origin:

- **Geometry to Origin**: Move the object so that its middle is now wherever the origin was located.
- **Origin to geometry**: Move the origin to the middle of the object. This is the most common option chosen.
- **Origin to 3D Cursor**: Relocate the origin to where the 3D cursor is.
- **Origin to Center of Mass**: Calculate the center of mass of the object and move the origin there.

Duplicating objects

Blender can duplicate existing objects. This is very useful and can speed up making things. To duplicate an object, follow these steps:

1. Select the object to be duplicated.
2. In the **3D** menu, choose **Object** | **Duplicate** or press _Shift + D_.
3. Move the mouse or use the arrow keys to place the duplicate (axis-constraining commands also work at this point).
4. Press _Enter_ or the select mouse button.
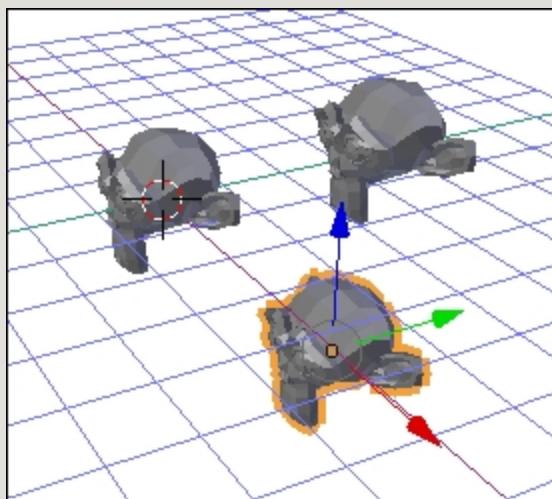
Object selection

Naturally, Blender is capable of selecting more than one object at a time. Blender has many tools to help when selecting objects. With multiple objects, the transformation commands work on all the selected objects at once. This is a powerful way of controlling transformations while keeping objects in relative positions to each other.

> **NOTE**
>
> To practice selection, there need to be multiple objects to select. Create a scene, add an object, and then use the duplicate (*Shift + D*) command to create many objects from the one. Spread them around the scene so that they can be selected with the following tools.

Shift select

One way to select multiple objects at once is to hold down the *Shift* key while clicking on the desired objects one at a time. To deselect an object, it must first be made active. The active object is highlighted in a different color. Then it can be clicked again while holding the *Shift* key, and it will be removed from the selection.

Border select

Another way to easily select multiple objects in a scene is to use the border select command and draw a box around the objects you want to select. To border select, choose **Select | Border Select from the 3D View** menu or press *B* on your keyboard. Then, click and hold down the select mouse button, move the mouse pointer, and release the mouse button:

(1) Suzanne.006
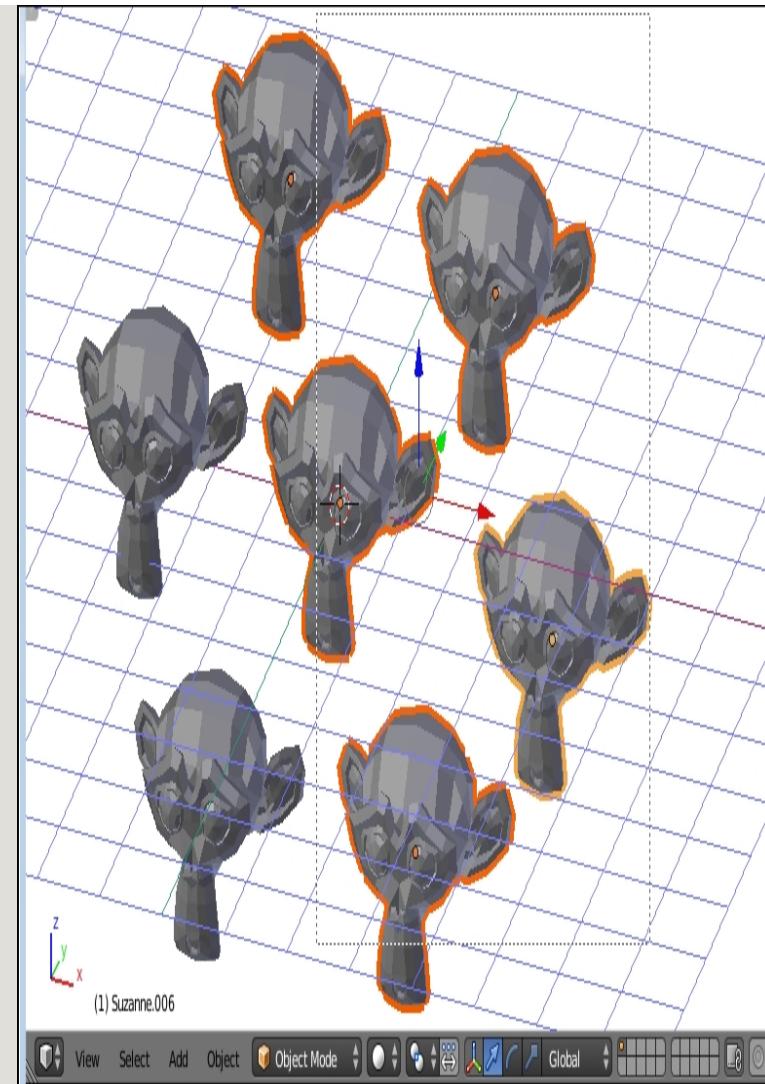
If even a small part of an object is included in the box, it will be added to the selection.

To deselect objects with border select, use the middle mouse button when drawing the box.

Circle select

Circle select is another powerful tool for selecting multiple objects. To circle select, choose **Select | Circle Select from the 3D View** menu or press *C* on the keyboard. A circle will appear around the mouse pointer. Use the scroll wheel or *+* and *–* on the Numpad to increase or decrease the size

of the circle. Then, click or click and hold the select mouse button, and everything inside the circle will be selected. Use the middle mouse button to deselect:
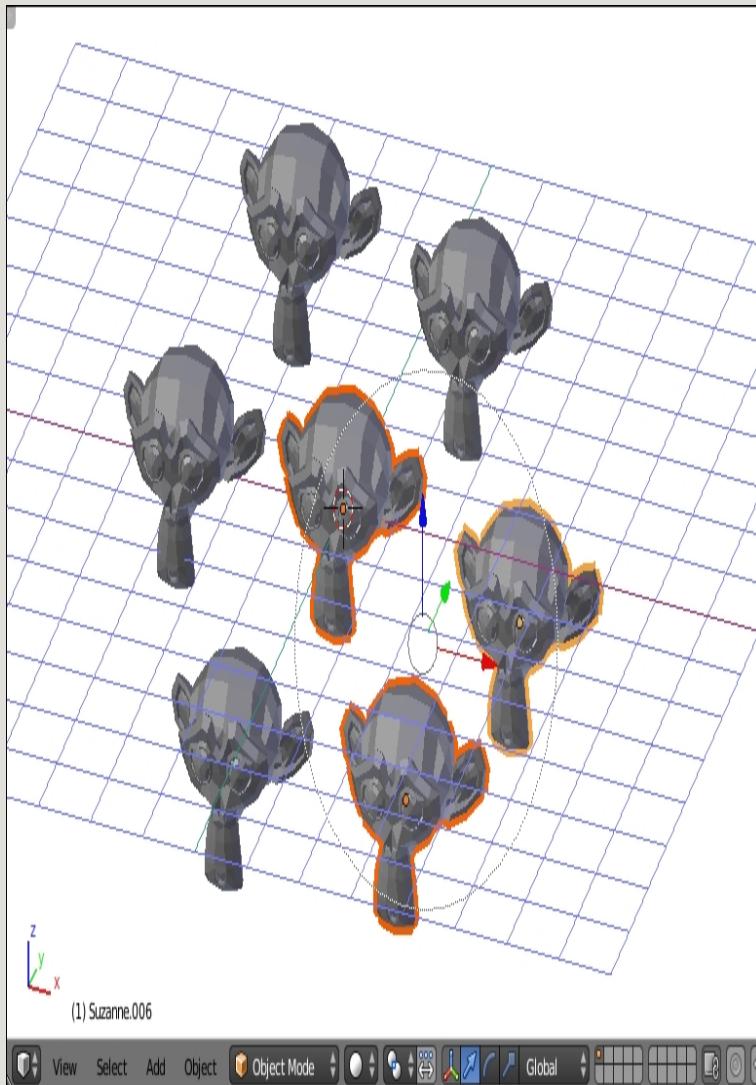
Circle select only adds or removes an object from the selection if its center or origin is inside the circle when selecting.

The Edit mode

In Blender, the **Edit** mode allows more access to the shape of a single object so that it can be manipulated in order to change its shape. To enter **Edit** mode, use this method:

1. Select an object.
2. In the **3D View** menu, locate the mode pop-up menu and select **Edit Mode** or press *Tab* on your keyboard.

In **Edit** mode, the **3D View** menu, **Tool Shelf**, and **Properties** all change, adding new functionality only available in **Edit** mode:
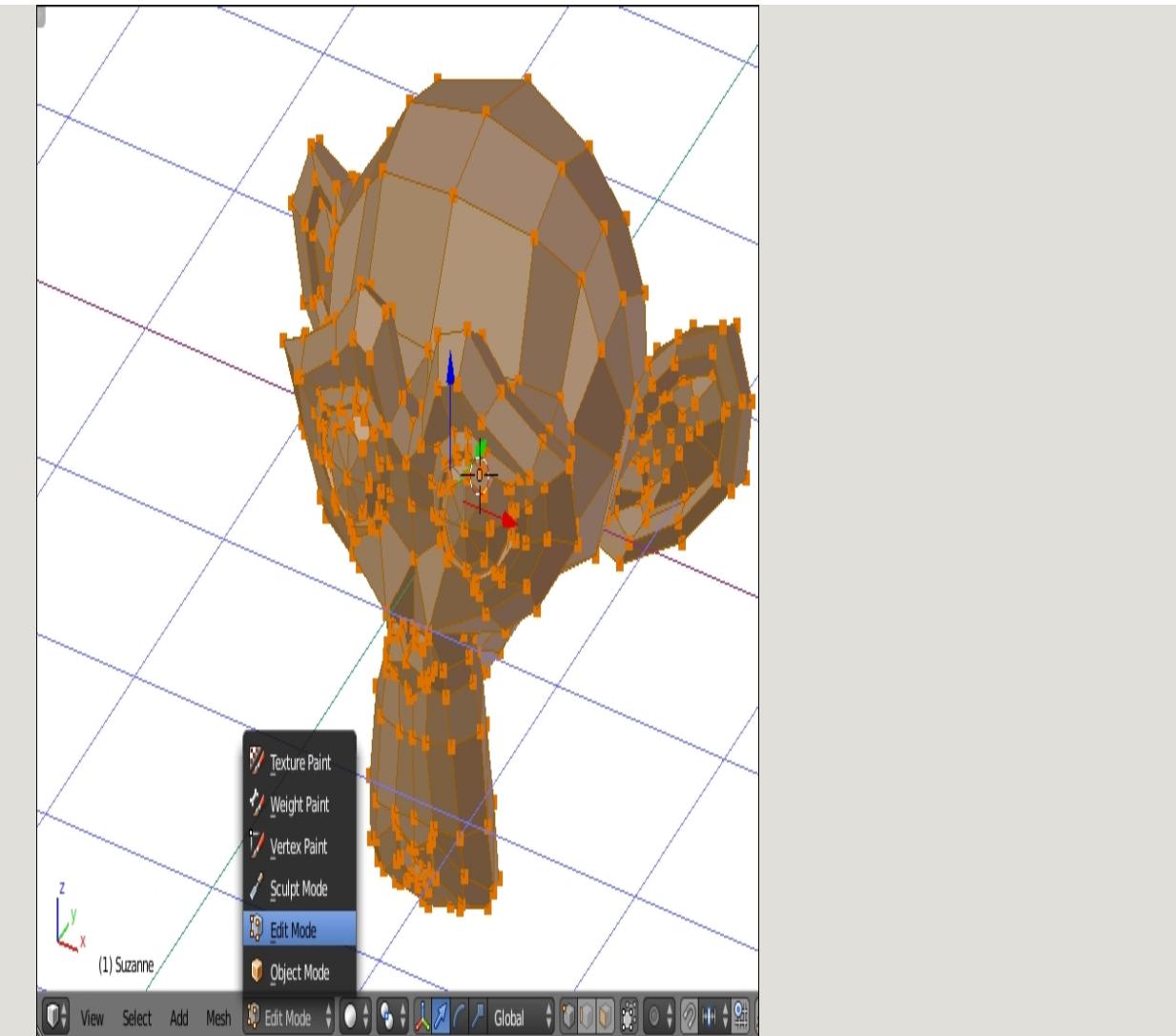
Parts of objects

In **Edit** mode, objects are broken down into three parts:

- **Vertices**: Points in three-dimensional space. Vertices don't have any shape by themselves
- **Lines**: Two points are connected with a straight line between them
- **Faces**: Three or more lines can be connected to make a face

There are many ways to think about vertices, lines, and faces. For instance, if making a kite, the vertices are the joints, the lines are the

sticks, and the bits of paper are the faces. If the location of the vertices is moved, the shape of the kite will change. It's the same with a 3D object. Editing by vertices, lines, or faces will affect the rest.

The **Edit** mode is by default in vertex select mode, meaning any selection or transformation is applied to vertices, but it's easy to switch by locating the vertex, line, or face select buttons in the **3D View** menu or by pressing *Ctrl + Tab* and selecting the desired option from the menu that pops up:
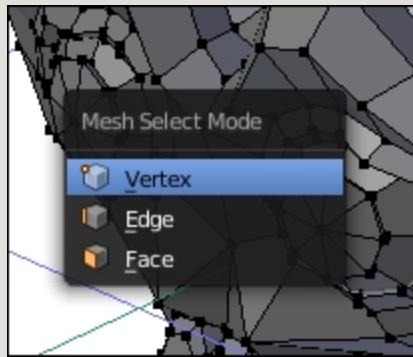
Try out **Select** mode. Select groups of vertices, edges, and faces, using circle and border select. Try moving, rotating, and scaling them and see how it effects the shape of the object. Try simple basic shapes such as cubes and circles. See how **Wireframe** mode affects selection. Notice how the other objects in the scene can't be accessed.

A lot of what happens in the projects will happen in **Edit** mode. But the individual projects will better teach you how to use it.

Incremental saving

It is always a good idea to save your work frequently. To save your work in Blender, choose **File** | **Save** from the info panel or press *Ctrl + S*. Then, navigate to a chosen folder or directory, give the file a name, and click on the **Save As Blender File** button or press the *Enter* key.

It is recommended that each Blender project gets its own folder and that all projects be saved in a location that will be easy to find later.

It is also a good idea, while learning especially, to give each version of the file you save a slightly different name. This way, there is a history of the work done and it's easy to go back in the case of a mistake that isn't discovered immediately. This is called incremental saving and is simple to do in Blender. Simply choose **File** | **Save As** or press *Ctrl + Shift + S* to get to the save menu. If the project has been saved previously, it should already have a name. Next to the filename, there are plus and minus buttons:
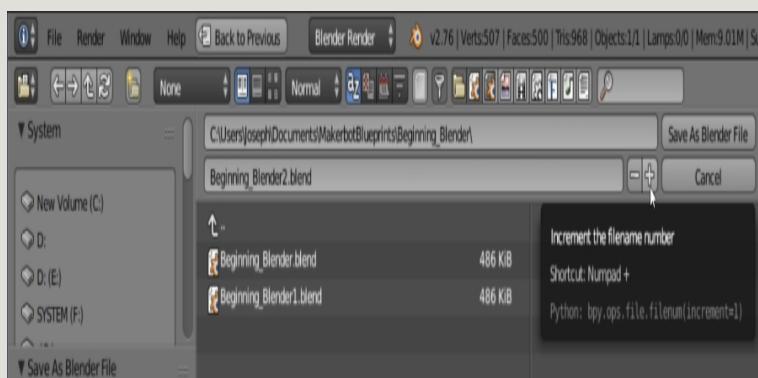
By clicking on those buttons, the filename will have a number attached to it, and that number will be increased every time the plus button is

clicked. Then, click on **Save as Blender File** or press the *Enter* key to save the file with a new filename.

Blender to real life

By default, Blender units don't make any attempt to relate to real-life measurements. However, after exporting a mesh, the slicing software will interpret the Blender units as millimeters, generally. So, it is good to think of Blender units as millimeters. This means that default objects in Blender are 2 mm across when they're added, which is fairly small.

Remember that one blender unit will be 1 mm when the file is used to 3D print an object.

Exporting an STL

Before they can be used in a 3D printer, 3D
models created in Blender have to be changed to
a file that the 3D printer can use. Blender's
default file format isn't readable by 3D printers
and can sometimes contain additional
information that the 3D printers don't need.

Most 3D printers use a file format called
STereoLithography (**STL**), which contains just
the final shape of the object. To export a finished
model to an STL for 3D printing, follow these
steps:

1. Select the model or models to be exported.
2. From the **Info** panel, choose **File** | **Export** | **Stl (.stl)**.
3. Navigate to a chosen folder or directory.
4. Give the file a name.
5. Click on the **Export STL** button or press the *Enter* key:
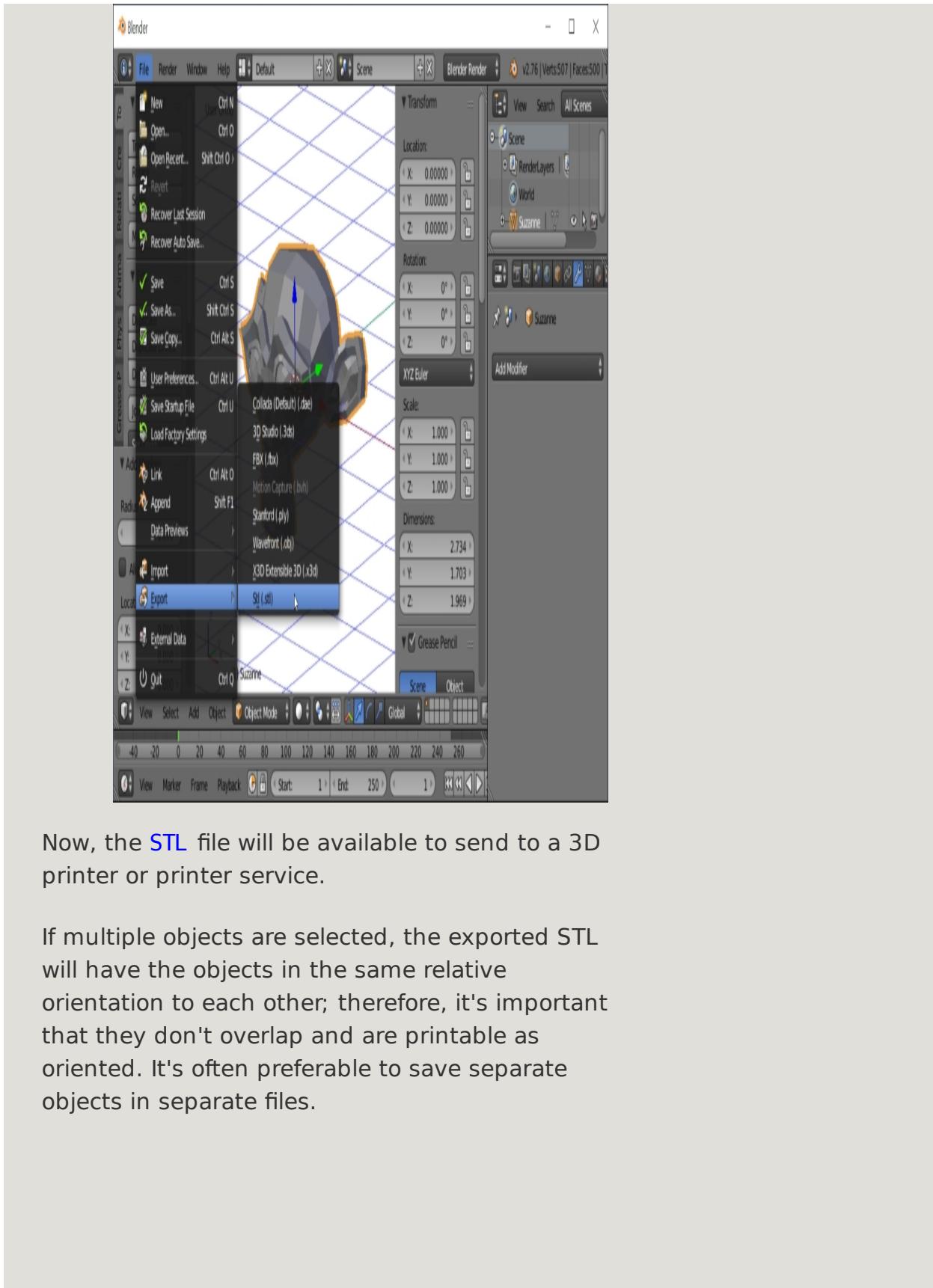
Now, the STL file will be available to send to a 3D printer or printer service.

If multiple objects are selected, the exported STL will have the objects in the same relative orientation to each other; therefore, it's important that they don't overlap and are printable as oriented. It's often preferable to save separate objects in separate files.

Summary

A recent poll of online social sites about 3D printing showed Blender was the most popular choice for creating models for 3D printing. The reason is obvious, taking into account Blender's vast functionality. However, it could just as well be that more people were talking about Blender because of its challenging learning curve.

Blender is capable of creating simple primitive shapes, viewing them from any angle, transforming them with precision, and manipulating their individual vertices, edges, and faces in the powerful editing mode. The model can then be exported to a file, ready to be 3D printed.

Blender has many functions not even covered in this chapter, such as sculpting, skeletal manipulation, and how to use individual modifiers to achieve specific results. As these functions become important for individual projects, they will be covered.

Hopefully, this chapter served to introduce how powerful and comprehensive Blender is. However, Blender's comprehensive nature comes at the cost of being complex, which can be overwhelming. But don't worry. Keep this chapter at hand to act as a reference and a crutch in future projects until Blender's functions become second nature. It usually only takes one project before Blender becomes second nature. Once Blender is familiar, no other 3D modelling software will be necessary.

Chapter 3. The Octopus Pencil Holder

3D printing makes it easy to combine form and function. Why have just a pencil holder when you can have a pencil holder that looks like, say, a cute octopus? This project is an excellent starting project because it demonstrates a simple but versatile modeling technique that involves editing a simple mesh and smoothing it to create more detail.

This project will involve simple selection techniques unique to **Edit** mode, modification commands in **Edit** mode, and applying modifiers to soften and combine shapes. This technique alone can be used to make an unlimited number of cool things once mastered.

In this chapter, we will cover:

- Planning the project
- The first basic shape
- Smoothing the mesh with modifiers
- Bending the tentacles
- Flattening the bottom
- Renaming objects
- Adding a pencil cup
- Adding a face
- Finishing touches

Planning the project

A pencil holder is basically a cup with a sturdy base that can be used to hold objects taller than itself, such as pens, pencils, and other items that

might otherwise clutter up a desk. A cute, cartoony octopus's tentacles will provide the perfect base and prevent the cup from tipping over.

The hole in the cup should be roughly cylindrical, about 40-50 mm wide at the base. The whole thing should stand at least 80 mm tall. And, of course, it needs a cute and friendly face, something like this:
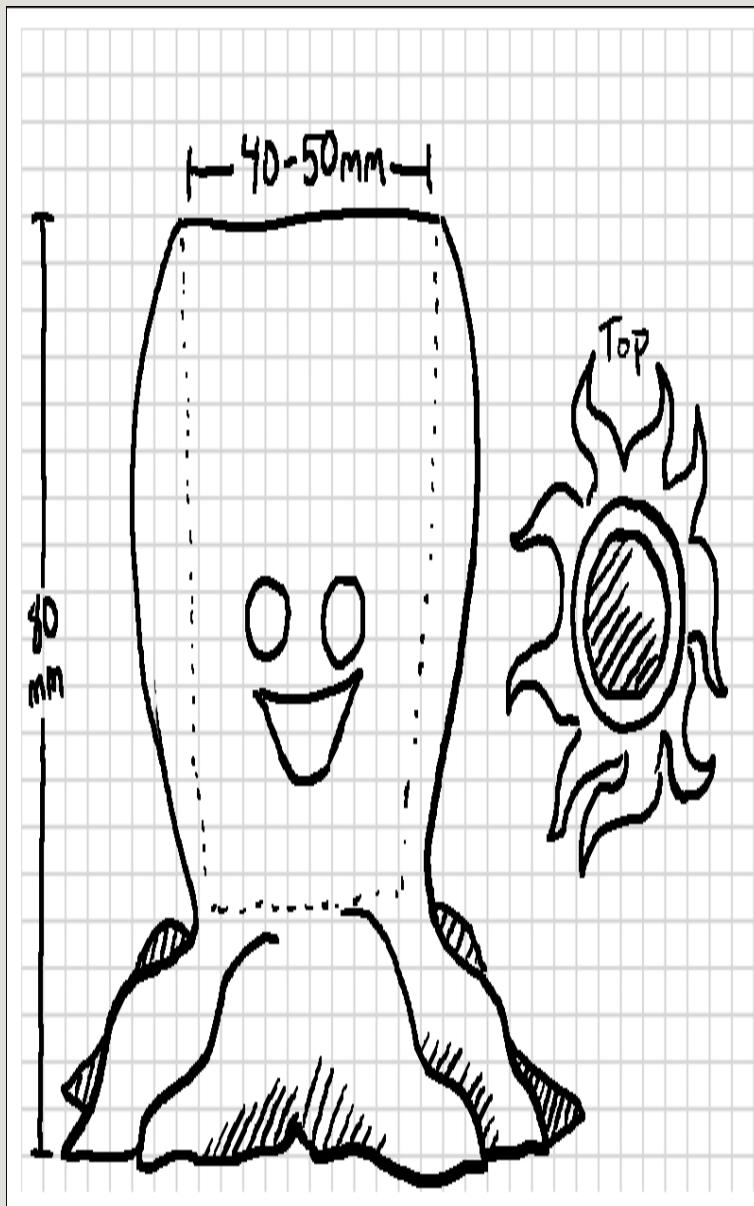
The first basic shape

This project is going to take advantage of several powerful editing tools that Bender provides. The first one is going to be the extrude operator. Extruding takes its name from a process for making things in real life, but in 3D modelling, extruding takes a selected part of an existing model and creates new geometry on the edge of the selected parts, so that the original can be moved away but remain attached to where it came from. The result is a new shape that can then be edited.

Let's get ready; it's time to get to work:

1. Open Blender, select all (*A*) in the default scene, and clear (*X*) it:
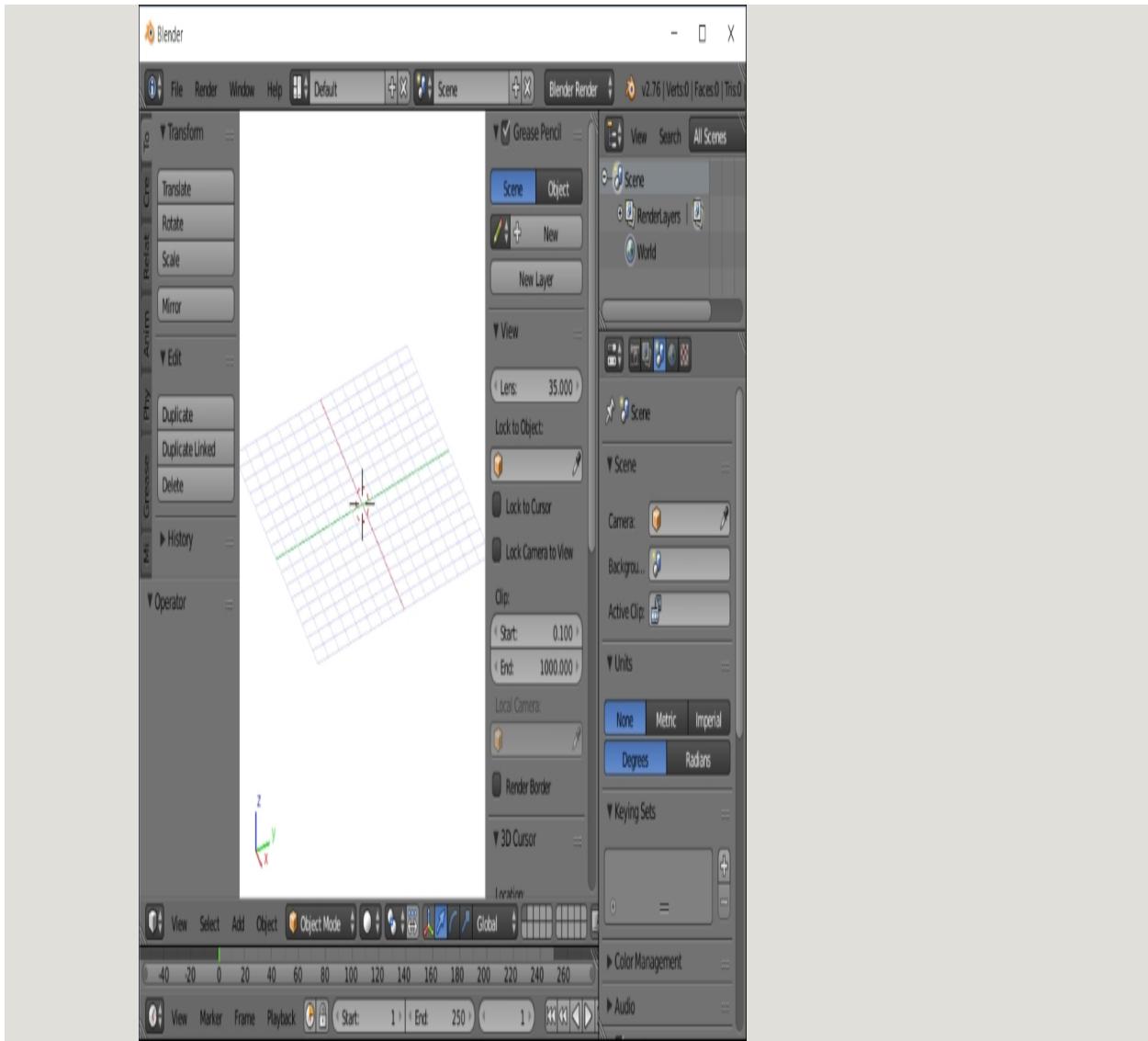
2. Next, add (*Shift + A*) a cylinder object to the scene:
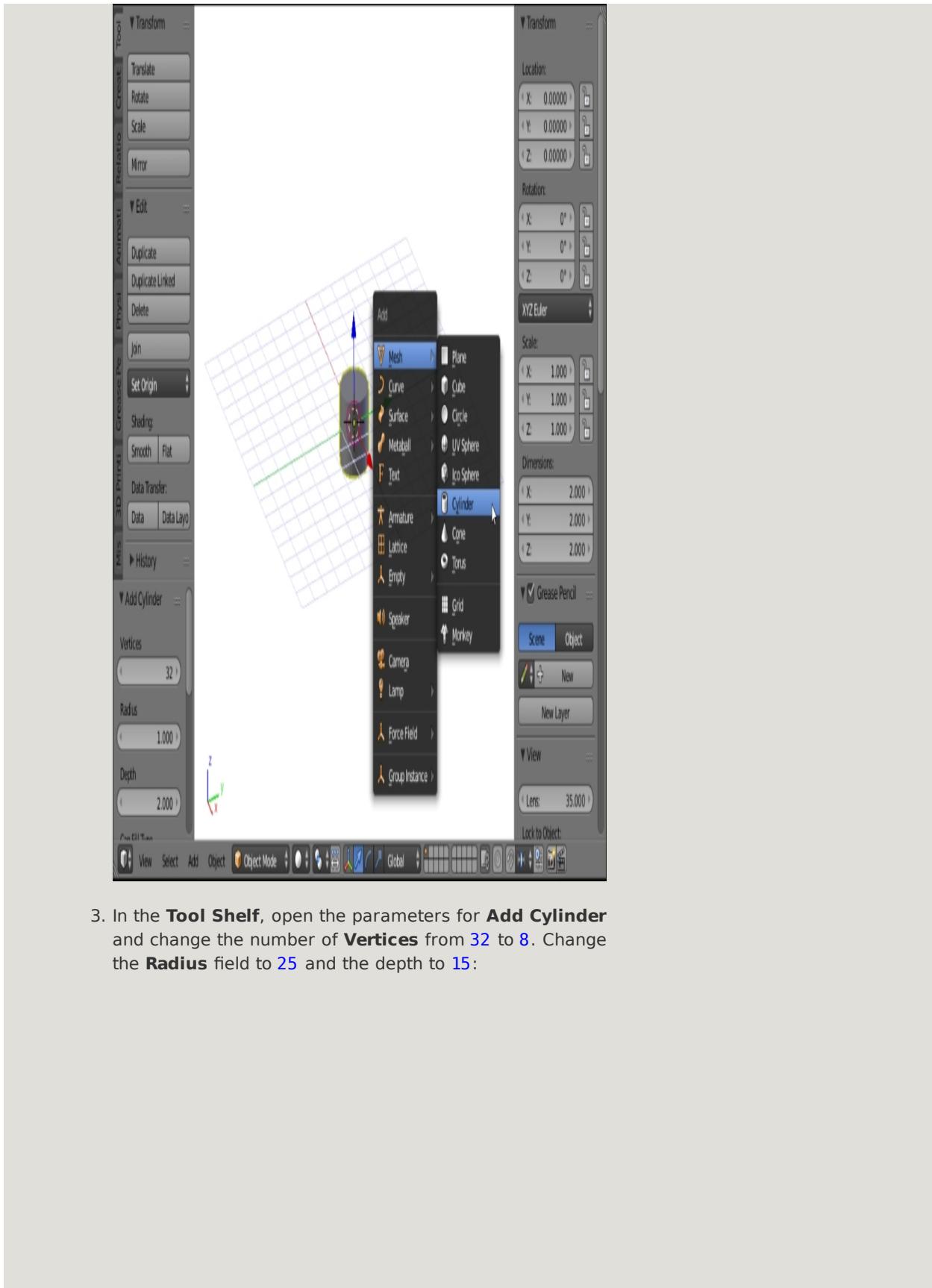
3. In the **Tool Shelf**, open the parameters for **Add Cylinder** and change the number of **Vertices** from 32 to 8. Change the **Radius** field to 25 and the depth to 15:
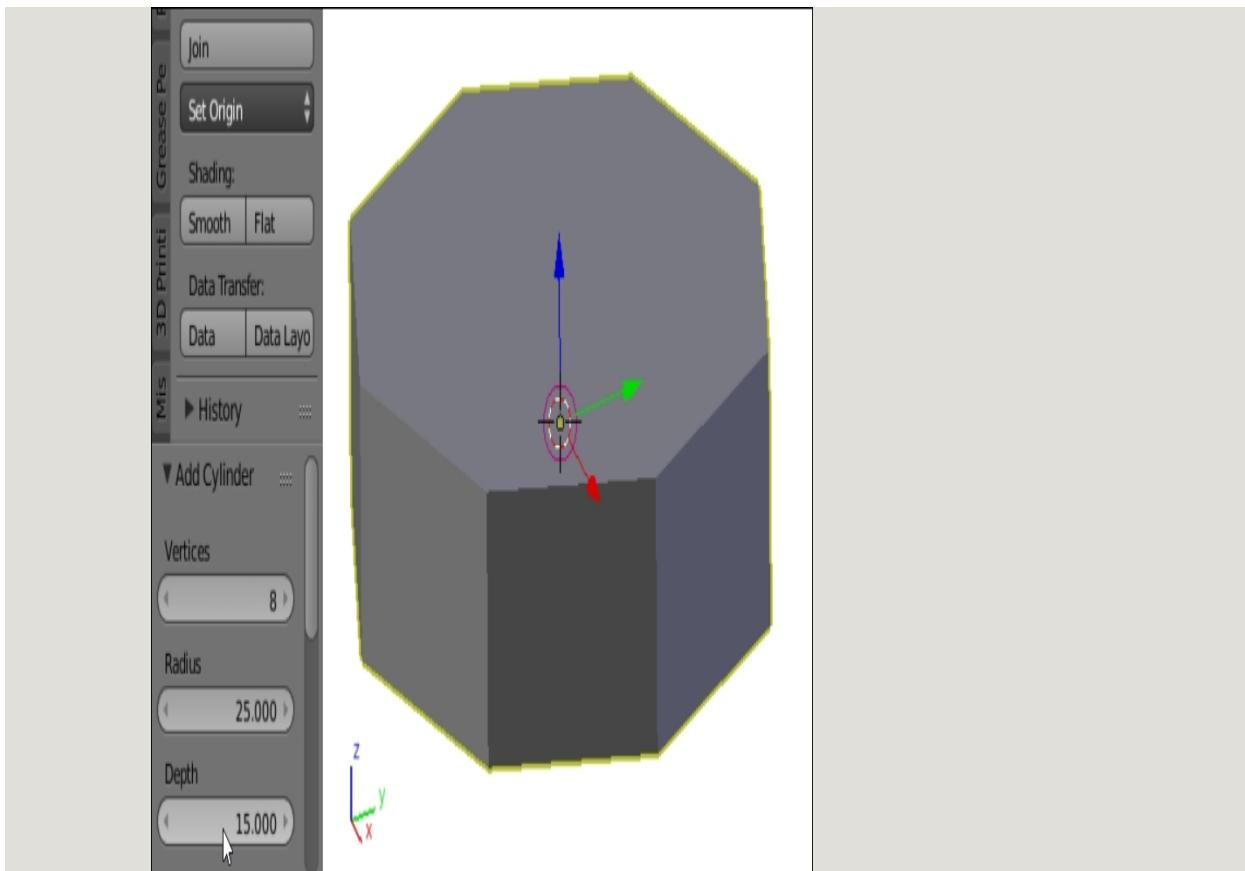
NOTE

Remember that these settings are only available when an object is first created. Selecting anything else, performing a transformation or action, or even deselecting the object will make these setting disappear. If the settings are gone, the only way to get them back is to delete and re-add the object to the scene.

4. Enter **Edit** mode (*Tab*) and switch to **Face Select** mode (*Ctrl + Tab*):
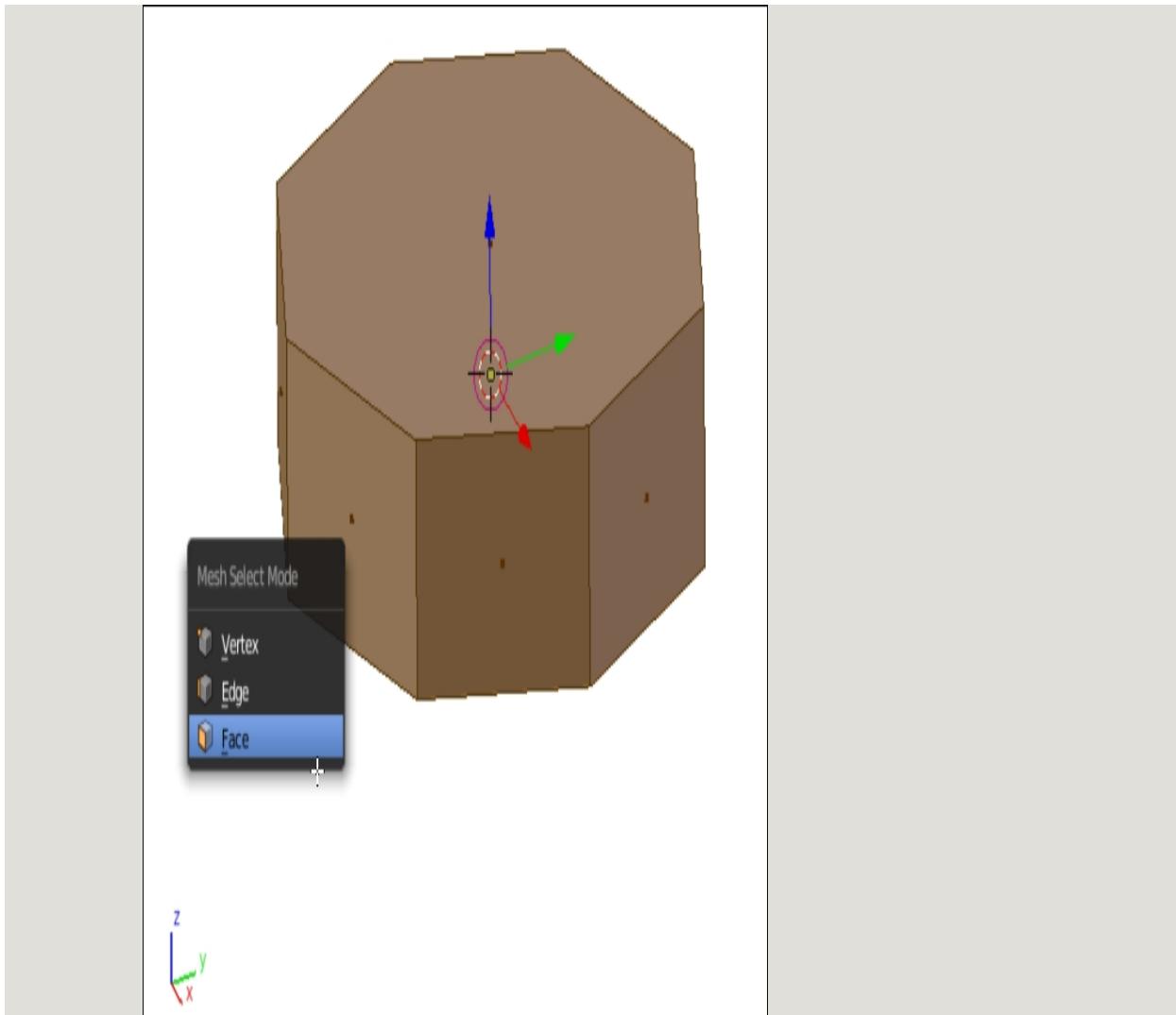
5. Deselect all faces (*A*).
6. Select one of the vertical sides of the cylinder.
7. In the **3D View** menu, navigate to **Mesh** | **Extrude** | **Region**, or press *E* on the keyboard.
8. Move the mouse until the leg is long enough for you to just type 40 to extrude it exactly 40 mm.
9. Press *Enter* or click on the select mouse button to complete the action:
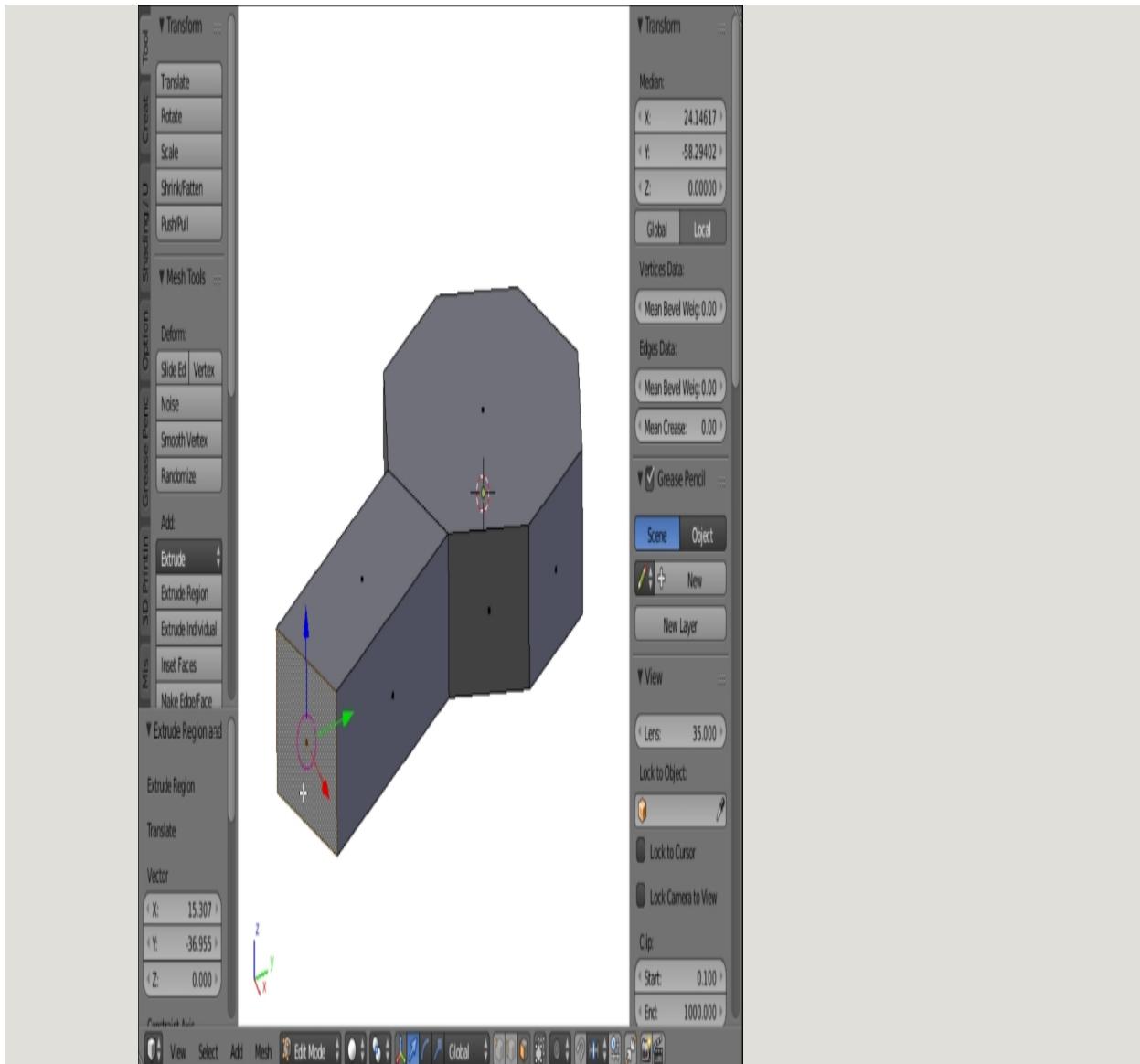
10. To give this arm a gentle, tentacle-like shape, scale the
    selected face at the end of the arm (*S*) down to about 20
    percent (*.* + *2*):
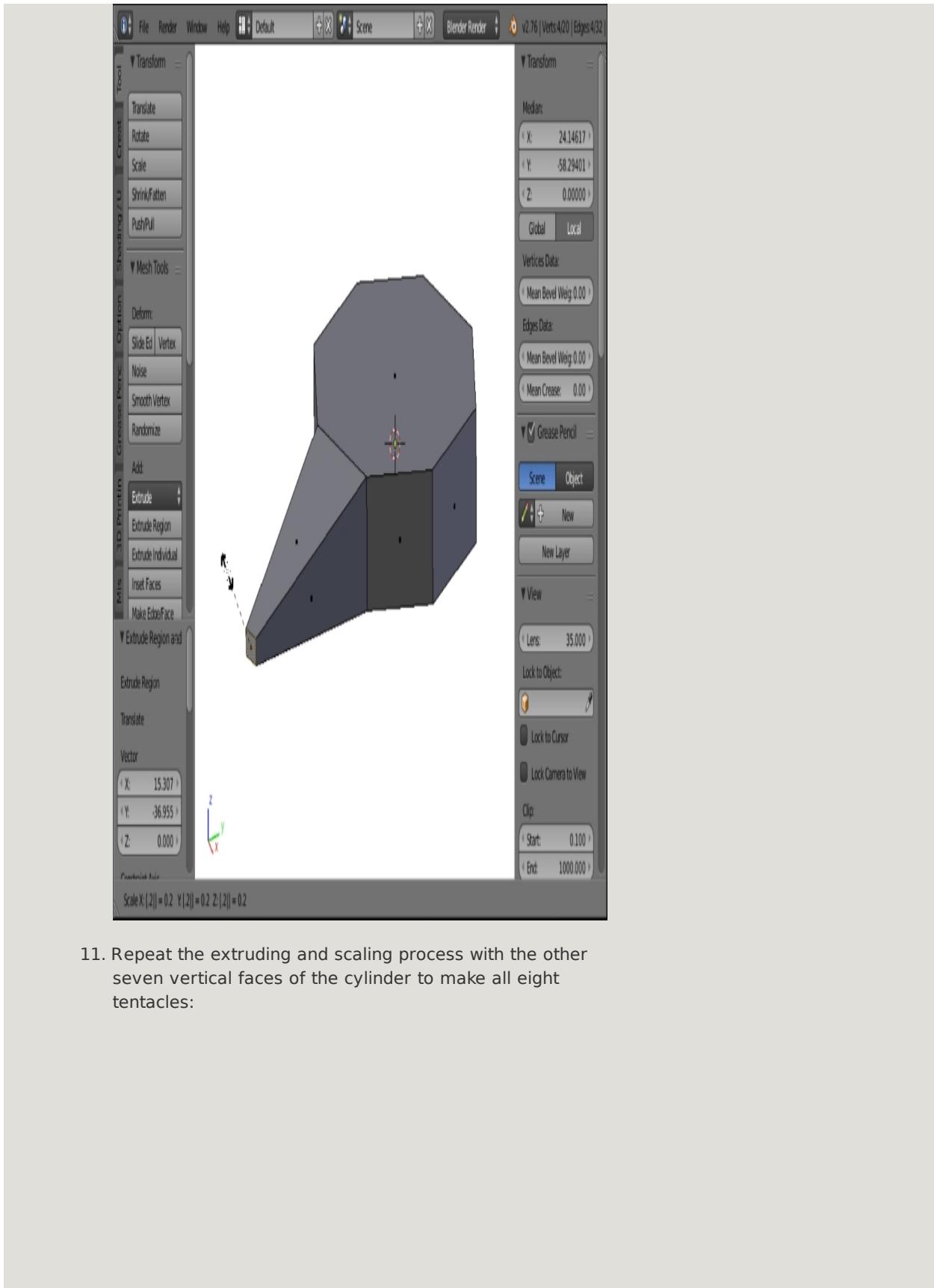
11. Repeat the extruding and scaling process with the other seven vertical faces of the cylinder to make all eight tentacles:
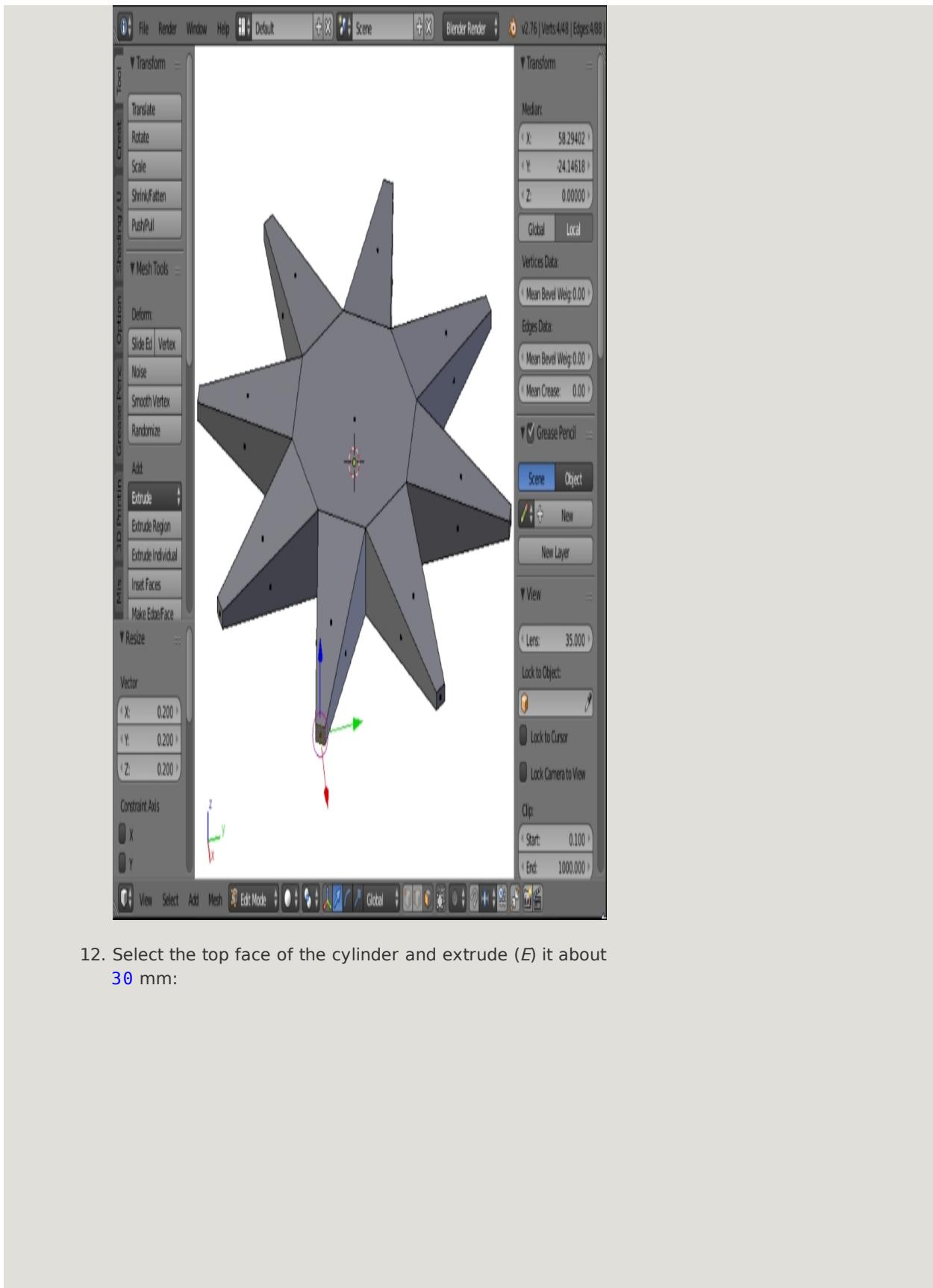
12. Select the top face of the cylinder and extrude (*E*) it about 30 mm:
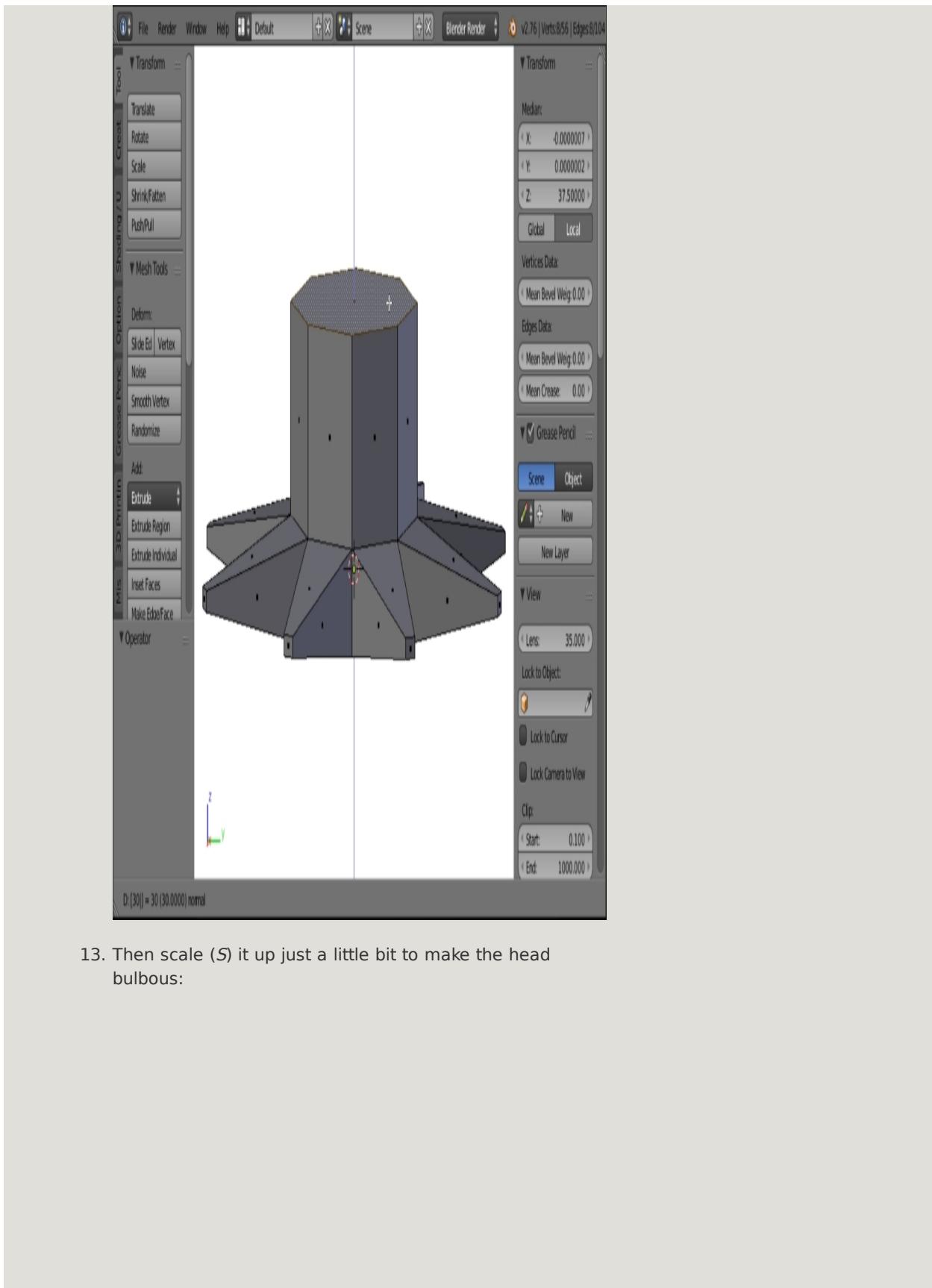
13. Then scale (*S*) it up just a little bit to make the head bulbous:
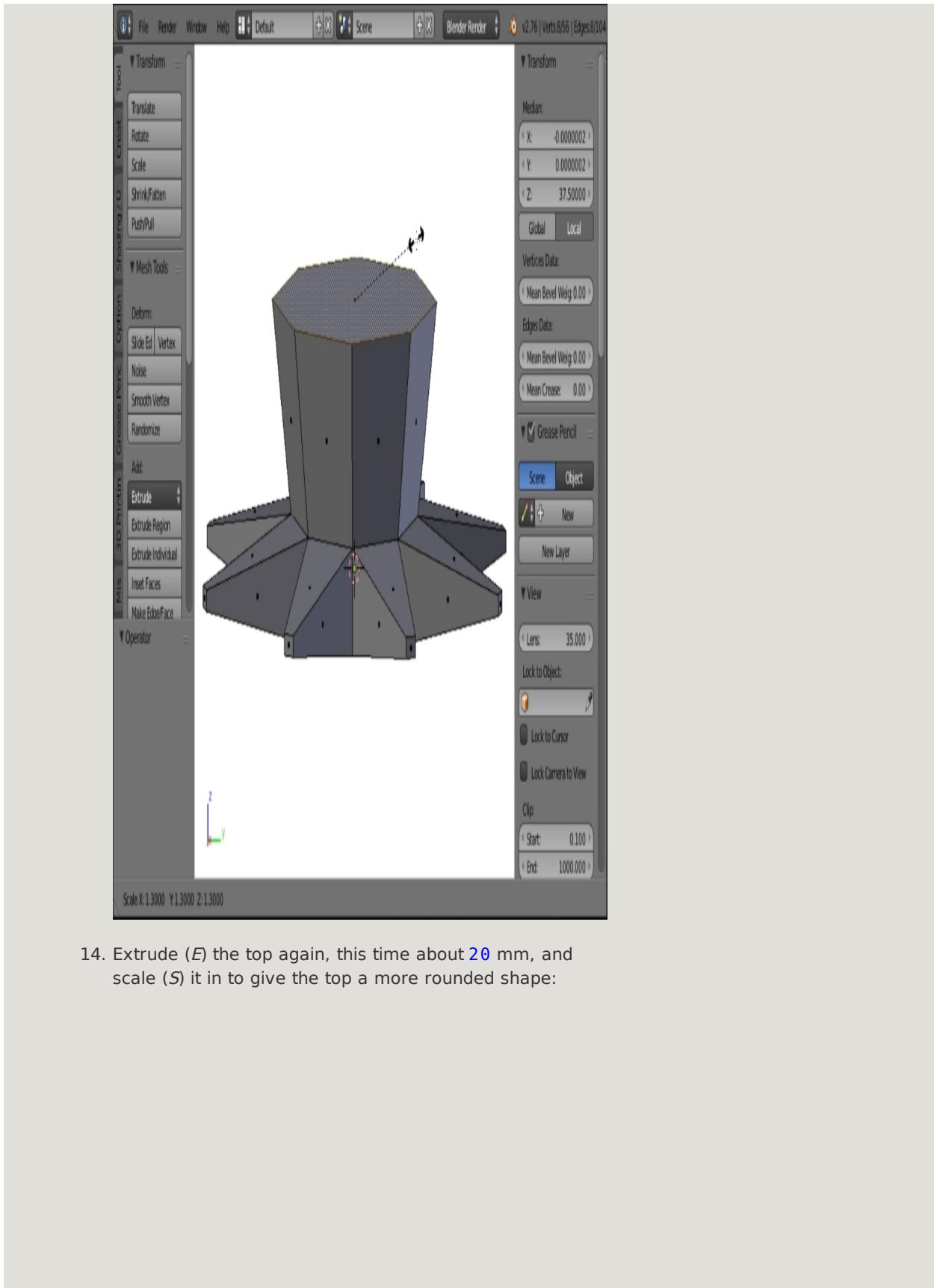
14. Extrude (*E*) the top again, this time about 20 mm, and scale (*S*) it in to give the top a more rounded shape:
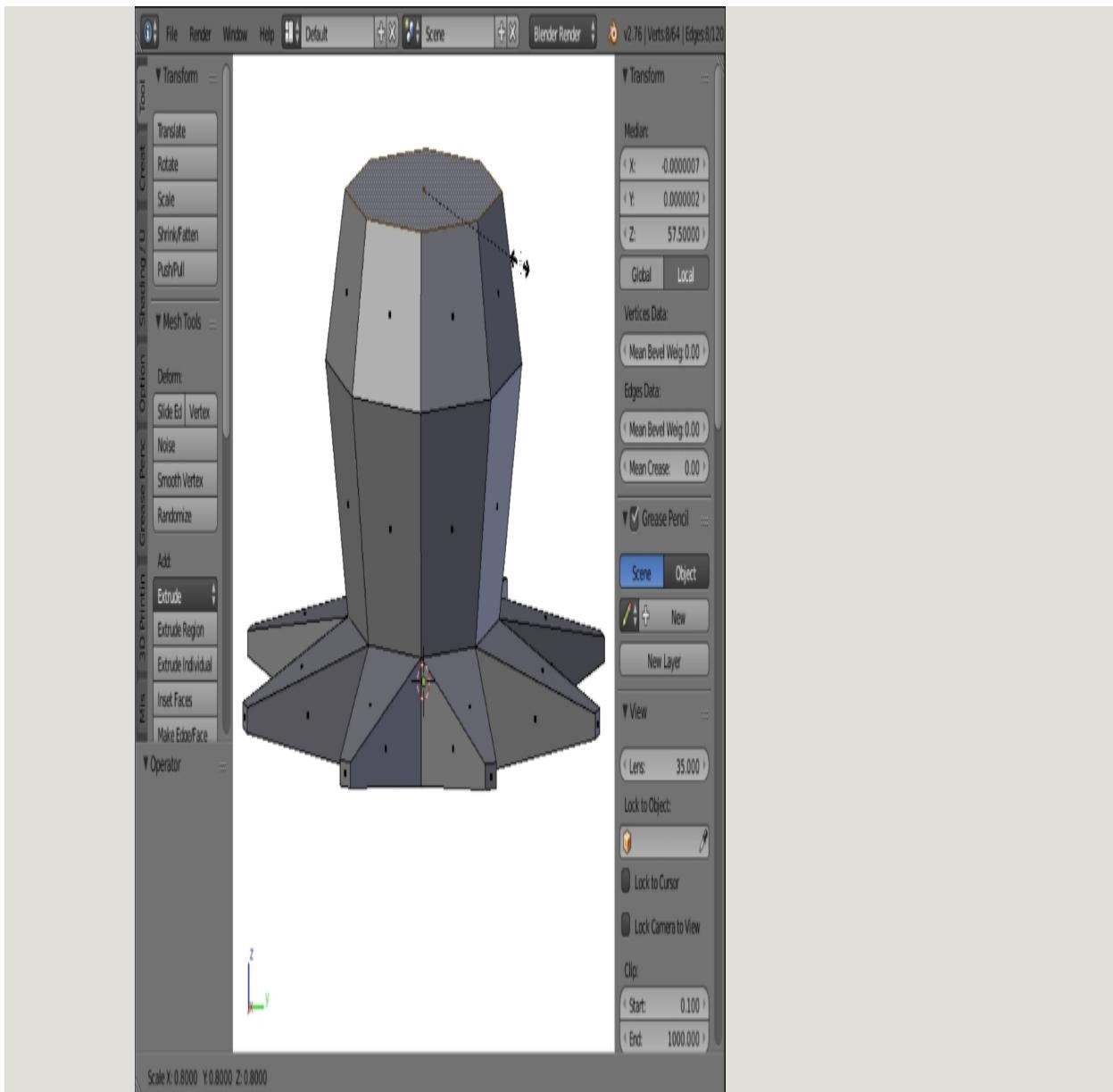
Like making a carving out of stone, these first stages are rough, but the model will be refined later. As it is, this basic shape has got a head and eight arms. That's enough like an octopus to get started.

Smoothing the mesh with modifiers

This blocky octopus is fine for editing; it doesn't have many parts to keep track of. But it needs to be much smoother for the final result. Fortunately, there's a way to increase the smoothness of the model while retaining a simple geometry that is easy to edit. This is done by adding a **Subdivision Surface** modifier to the object.

1. To add a **Subdivision Surface** modifier, click on the **Modifiers** tab in the **Properties** panel (the one that looks like a wrench):
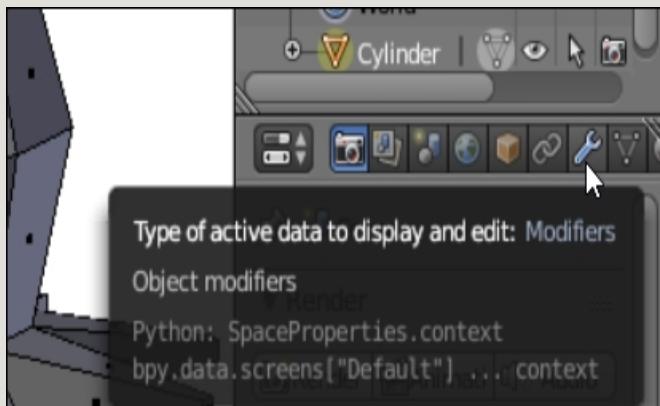
2. Click on the **Add Modifier** button, and in the menu that appears, choose **Subdivision Surface** from the list:
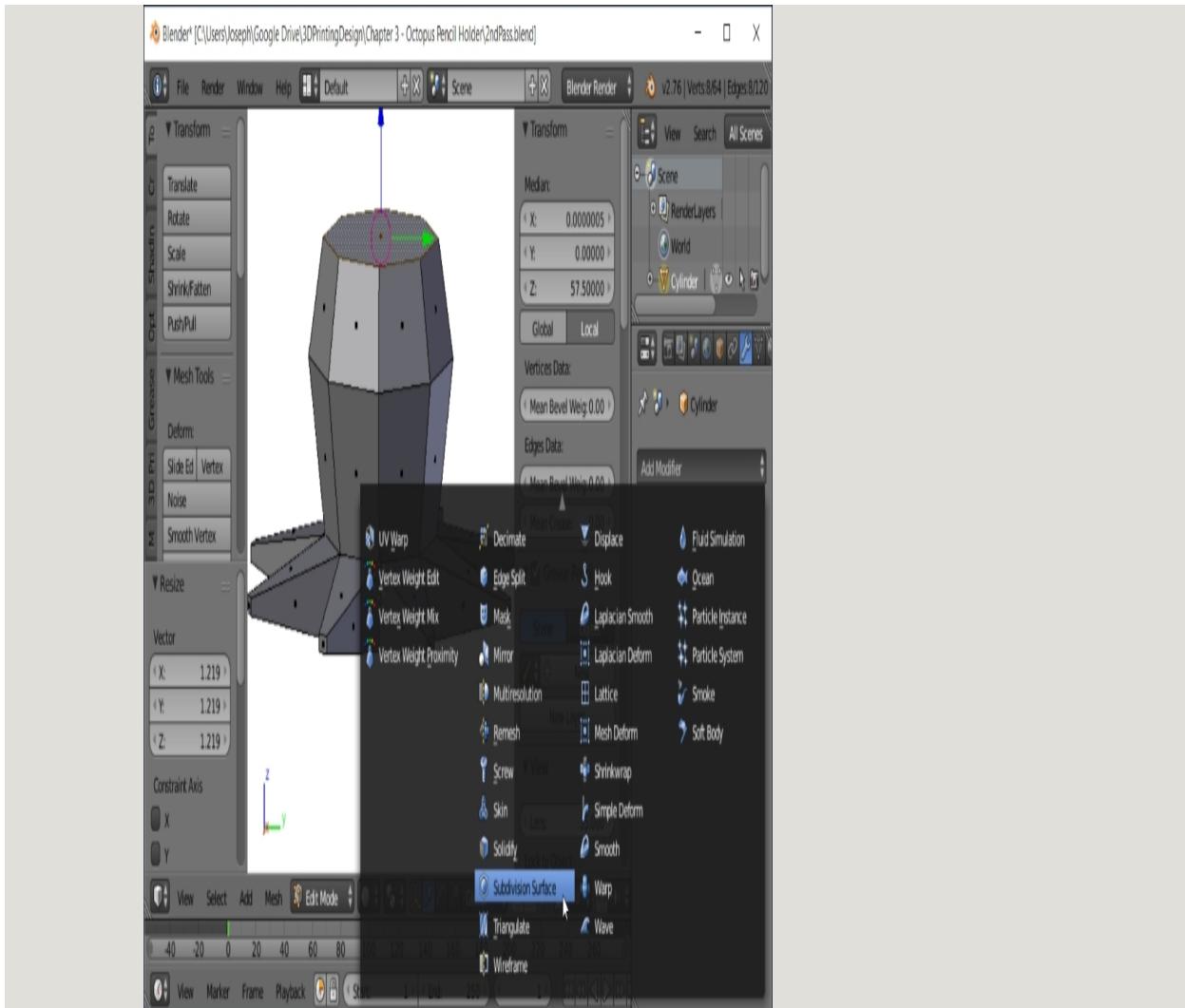
With the **Subdivision Surface** modifier on, the shape looks much smoother. While in **Edit** mode, it's clear that the original geometry is still there and acts as a sort of cage that defines the shape of the smoothed mesh. As long as the modifier isn't applied, the simpler geometry can be retained for editing.

TIP

Changing the **View** setting in the modifier will affect how smooth the mesh will be. Higher values will create more complex geometry that will look smoother but will slow down the computer more. There comes a point where increasing this setting won't make any noticeable difference. Generally, it's best to keep this setting high enough to have a good effect on the shape but low enough that it doesn't slow down the computer.

For now, set the **View** value to 2. This will give you something like this:
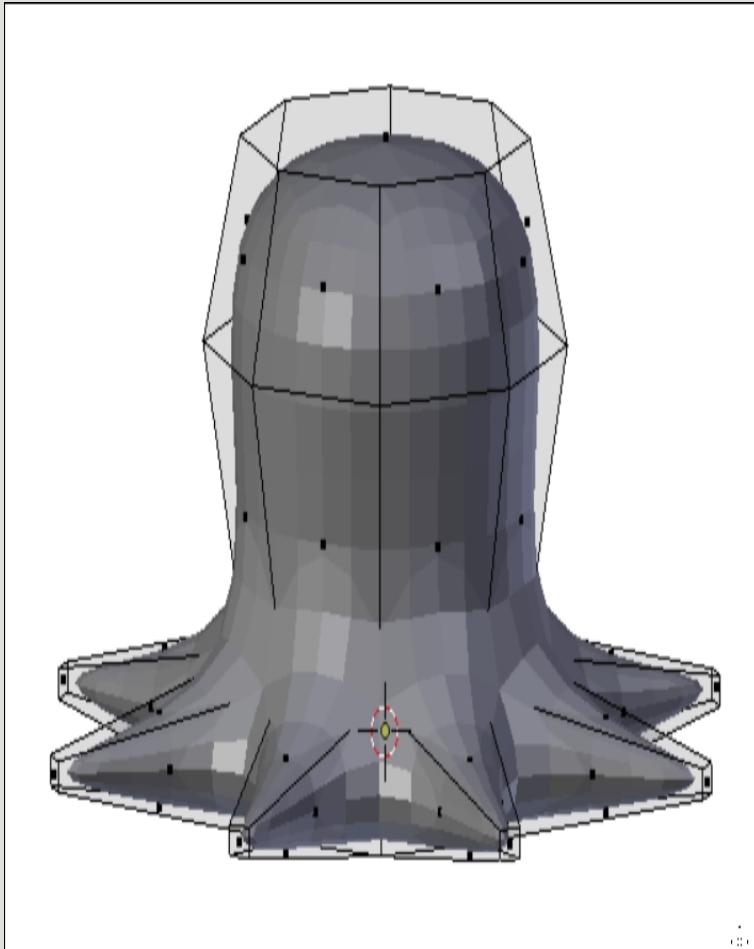
This isn't the final form, but it will get better. If seeing the smoothed mesh while working on the simpler mesh is confusing, the modifier can be temporarily turned off by clicking on its eyeball icon.

Bending the tentacles

Time to begin adding some details to the tentacles and give the model some personality.

Before moving ahead, let's take a look at another powerful tool for editing meshes called loop cut or loop subdivide, which will be used when bending the tentacles. Loop cut adds points in the middle of an edge and all around a portion of the geometry. These points can then be transformed. Loop cut only works with edge or **Vertex Select** mode and will automatically switch modes when being used.

For our project, we'll be using loop cuts to bend the tentacles. Follow these steps:

1. While still in **Edit** mode, switch to **Vertex Edit** mode (*Ctrl + Tab*).
2. Begin the loop cut operation by selecting **Loop Subdivide** under **Mesh | Edges** from the **3D View** menu or pressing *Ctrl + R*.
3. Move the mouse over the edge of one of the tentacles. Notice the preview line showing where the loop cut will be:
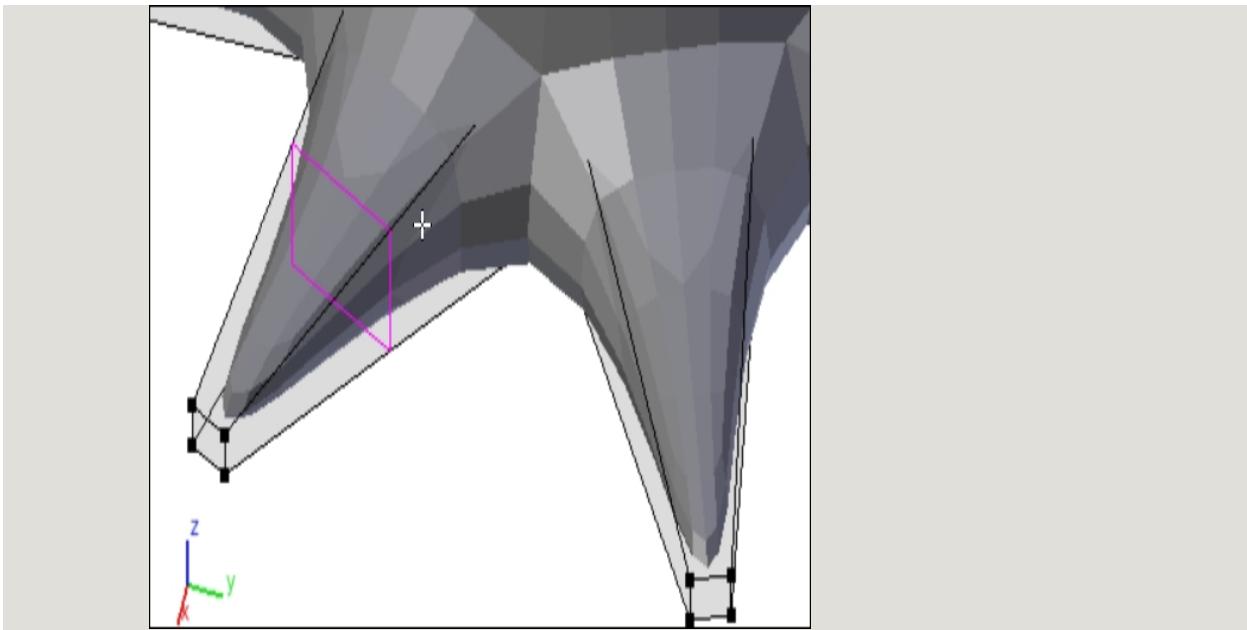
4. When the loop cut is where desired, click the select mouse button or press *Enter*. At this point, the loop isn't set and can still be slid back and forth by moving the mouse.
5. Try to keep this loop cut near the middle of the tentacle.
6. When it is where desired, press *Enter* or the select mouse button again to finish the operation.
7. After setting the loop cut, change the operator settings at the bottom of the **Tool Box** and change **Number of Cuts** to 2:
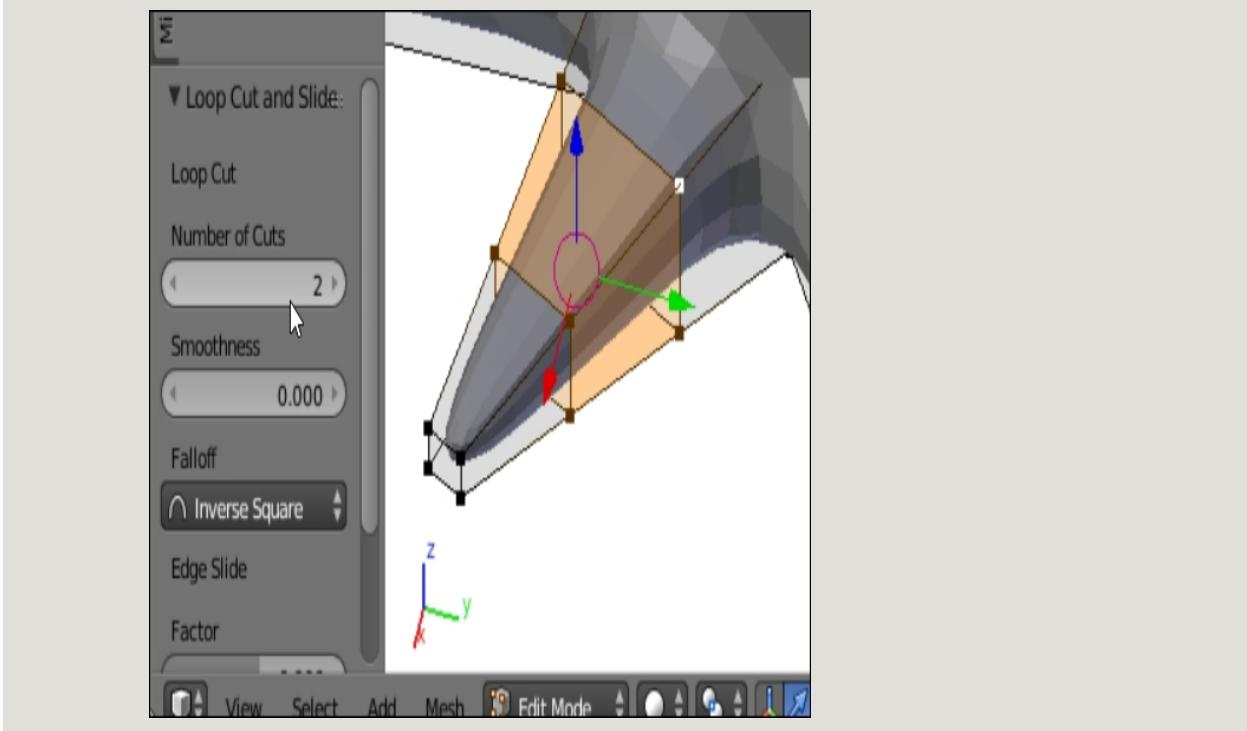
8. Deselect all points (*A*).
9. Select the points at the tip of the tentacle using either box (*B*) or circle (*C*) select.
10. Rotate (*R*) the points around the Z axis (*Z*).
11. Move (*G*) them along the X and Y axes (*Shift* + *Z*) to bend the tentacle a bit:
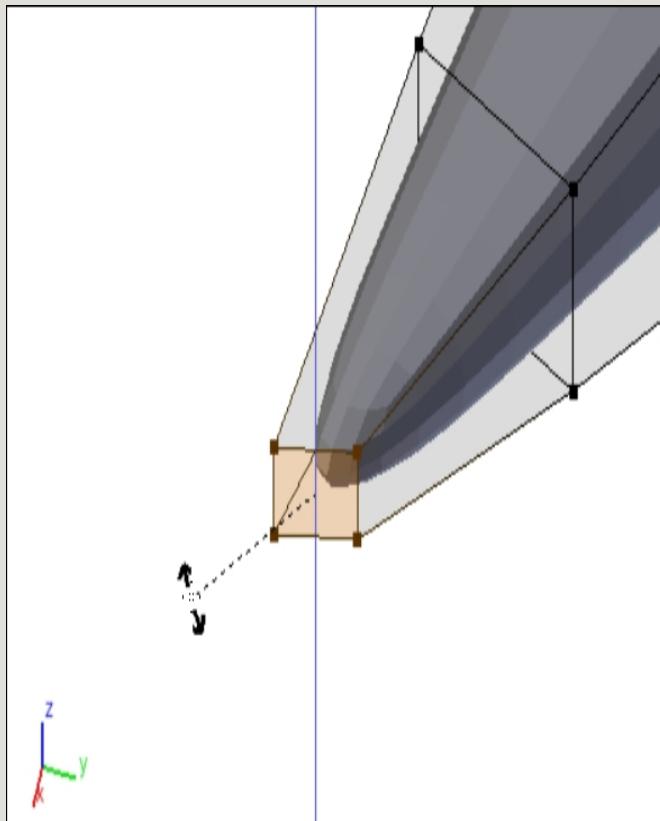
12. Expand the selection (*Ctrl* + Numpad *+*).
13. Again, rotate (*R*) the selection around the Z axis (*Z*) and move (*G*) it along the X and Y axes (*Shift* + *Z*) to bend the tentacle a bit more:
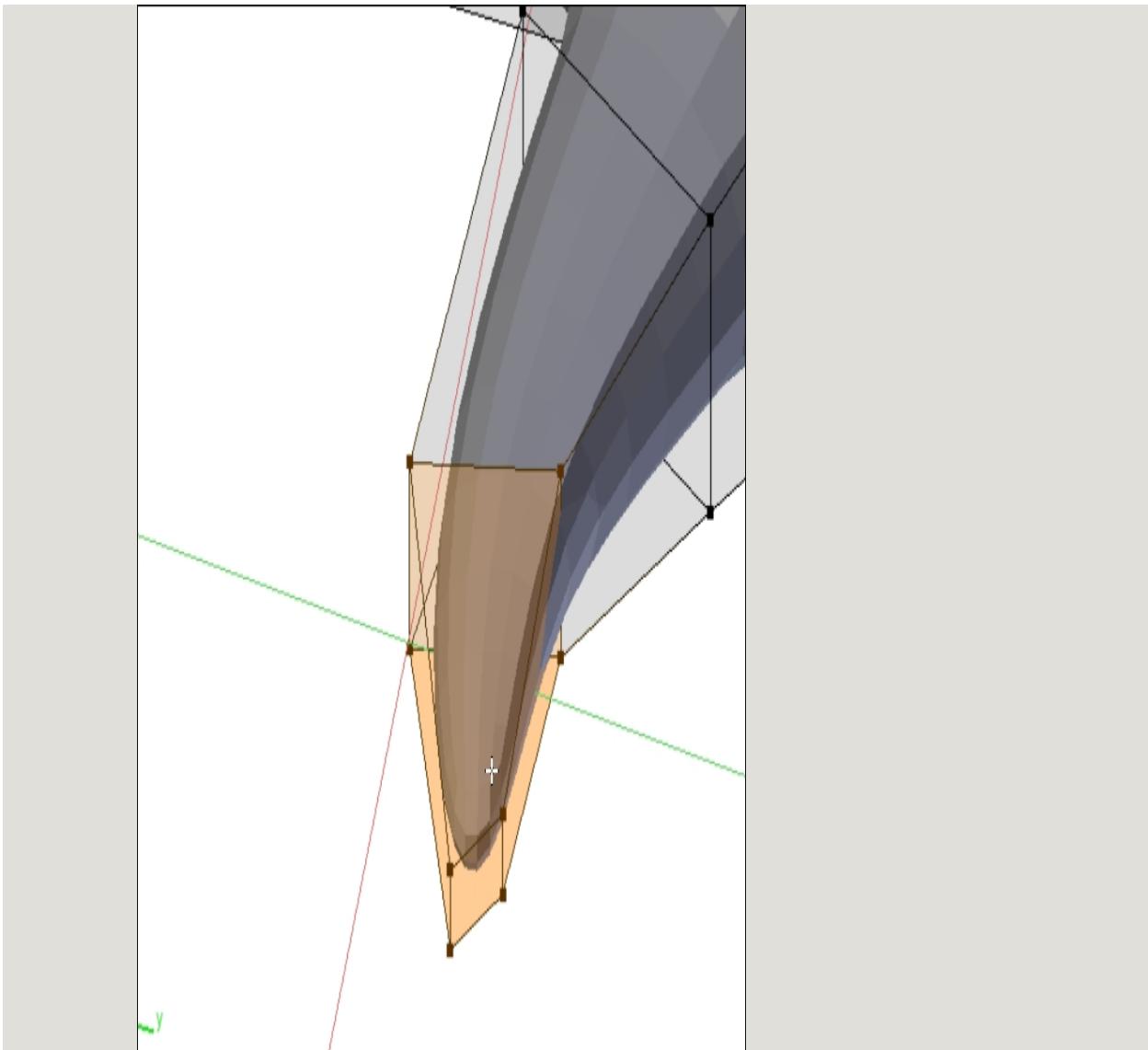
14. Expand the selection again. Rotate (*R*) and move (*G*) the tentacle, being careful to constrain the movement as before:
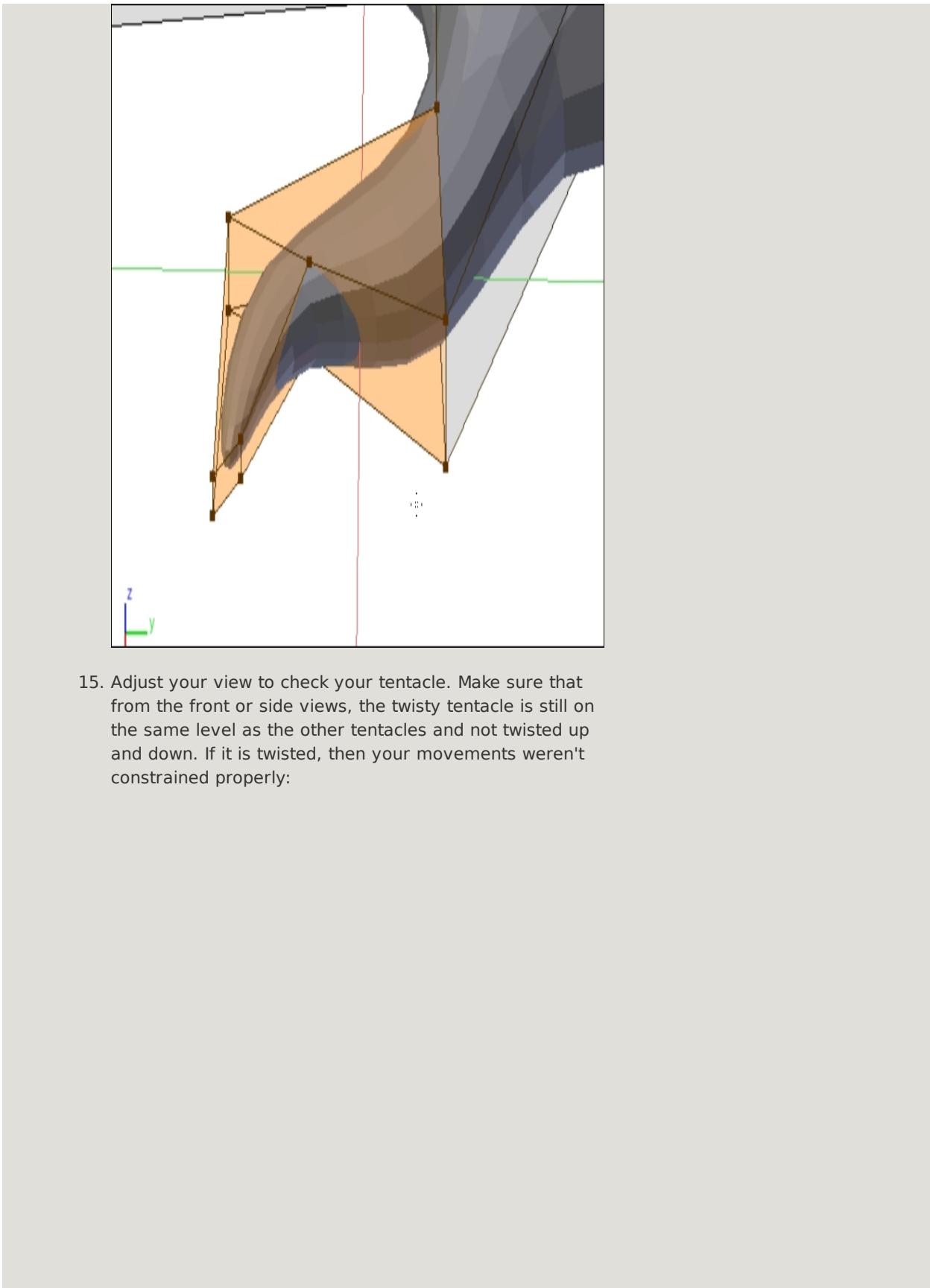
15. Adjust your view to check your tentacle. Make sure that from the front or side views, the twisty tentacle is still on the same level as the other tentacles and not twisted up and down. If it is twisted, then your movements weren't constrained properly:
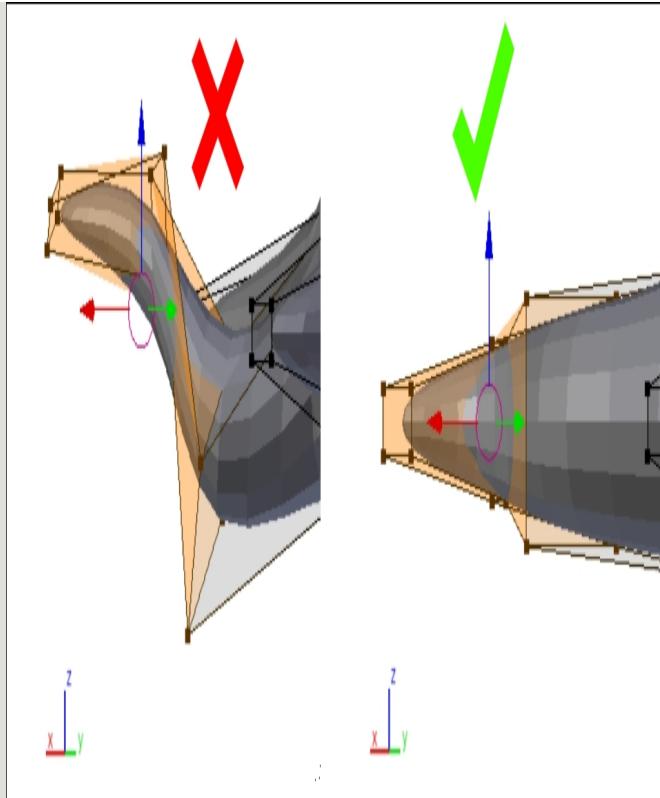
16. If it is not correct, then undo (*Ctrl* + *Z*) your movements and try again.
17. Another way to ensure that the movements are constrained properly is by performing all your transformations while in the top view (Numpad *7*). The trick to this method is to select points in **Wireframe** view (Z) to get all the points, not just the ones on top.
18. Move to the next tentacle and loop cut it. This time, while the loop location is being display, before clicking the mouse or pressing *Enter* the first time, try pressing the *2* key to quickly change the number of cuts to 2. This is a shortcut method that can optionally be used instead of changing the parameters after the cut:
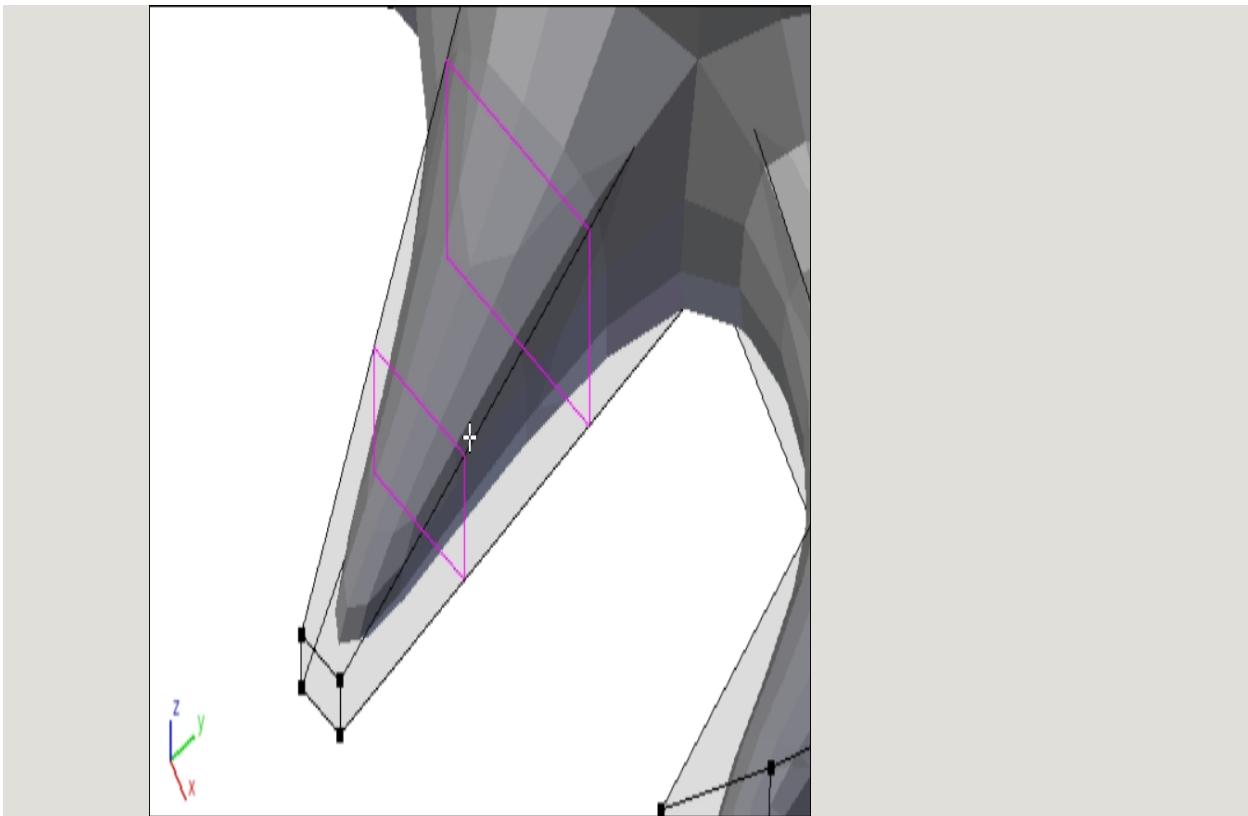
Work around the body, cutting, selecting, moving, and rotating each tentacle. Give each one a different twist, being careful not to overlap them. If the tentacles overlap, the model won't print properly when exported.
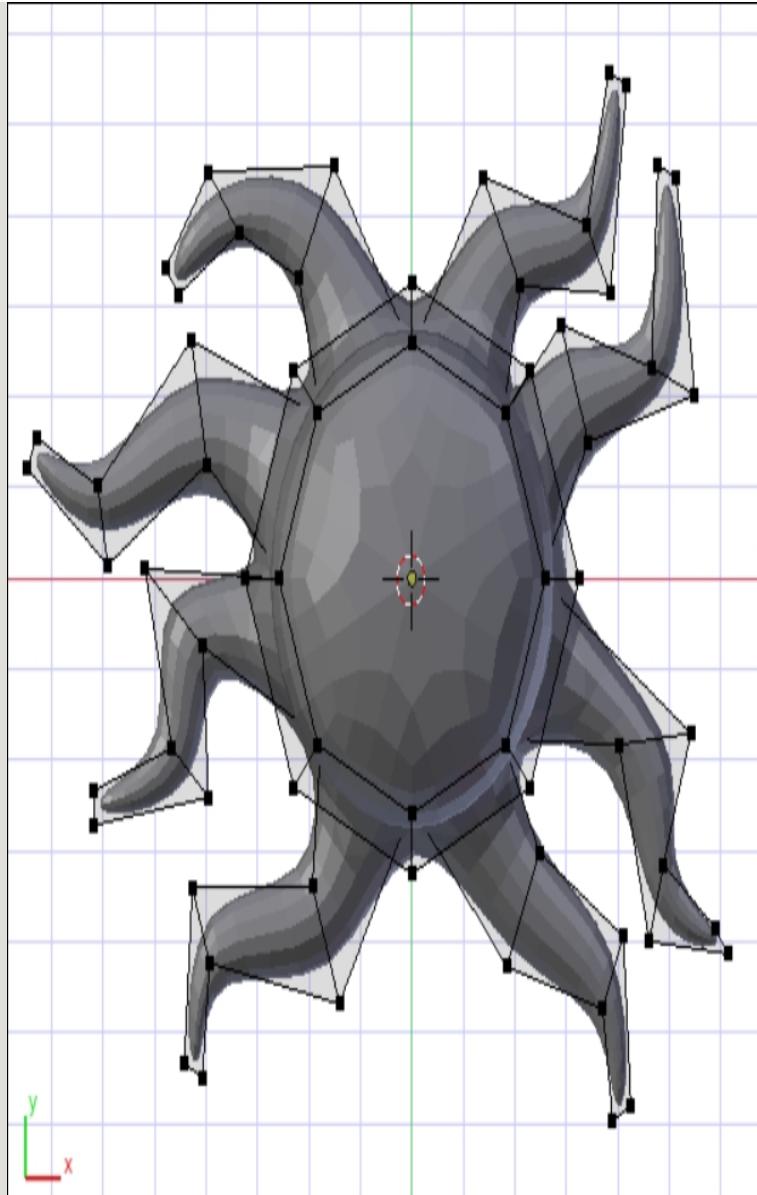
Your octopus doesn't need to look exactly like this. Make it your own. When all the tentacles have been given details, it should look much more like an octopus.

TIP

Some 3D printers won't be able to print this model if it's too big, so keep those tentacles tucked in and don't let them fly too far. The smaller 3D printers' build areas are about 150 mm or 6 inches squared. If the total area of your model is more than this, you'll need to think about the printer you're designing this for.

Flattening the bottom

An FFF-friendly 3D model needs to have a solid, flat base. There is more than one way the bottom of a model can be flattened. For one, the geometry can be edited to be flat. Alternatively, a floor object can be created and cut out of the object using a `Boolean` modifier. Since this project is using the **Subdivision Surface** modifier, editing the geometry can be complicated, so the floor method will be used.

Looking at the model from the front **Ortho** view (Numpad *1* and Numpad *5*), it's clear that the parts of the tentacles that sit below the X and Y axes (the red or green line, depending on the view) is where the model needs to be cut off to make a flat, printable base:
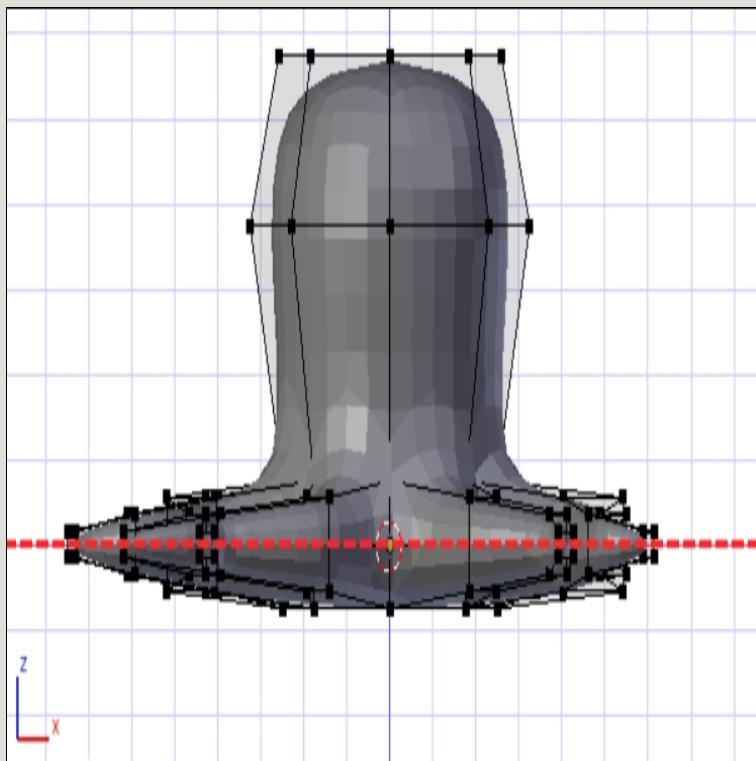
So, let's get started:

1. First, exit **Edit** mode (*Tab*).
2. Be sure the 3D cursor is at the **3D View**'s origin point (*Shift + C*). Then create a cube (*Shift + A*) by navigating to **Mesh | Cube**:
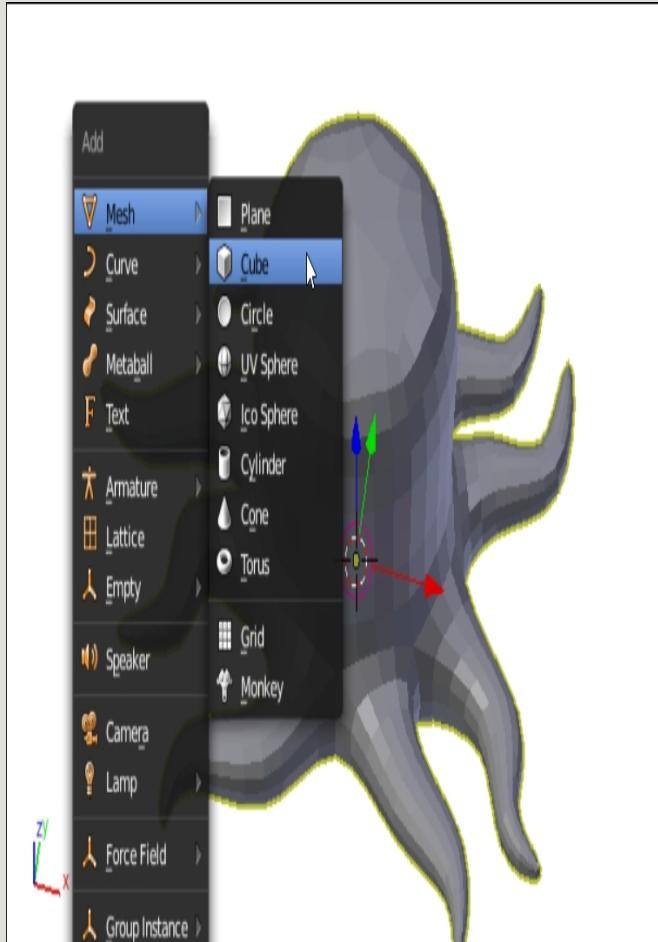
The cube cannot be seen in solid view because it's completely inside the octopus, so with the cube still selected, switch to local view (Numpad slash /) to view the cube alone:
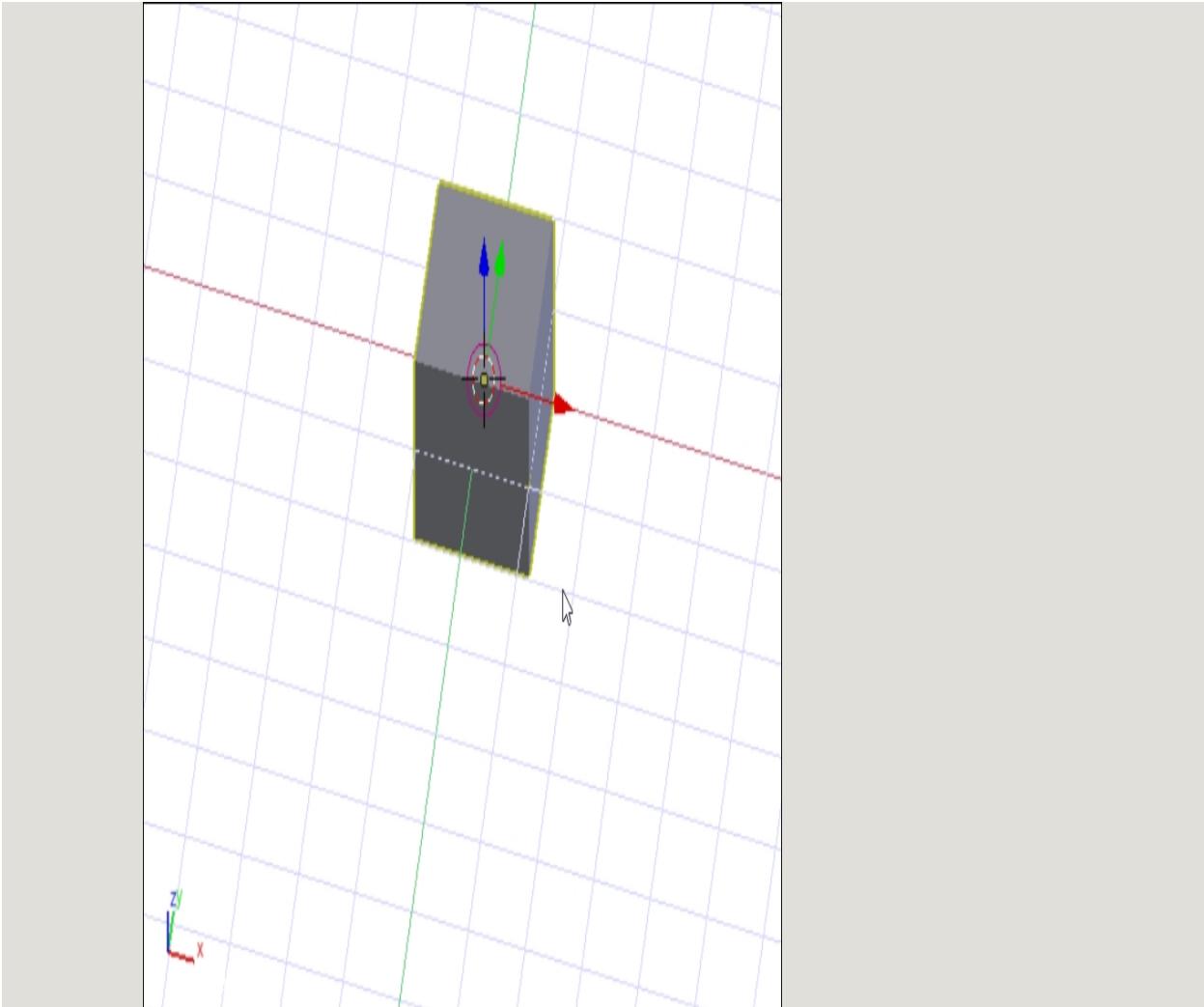
The cube is half above and half below the origin. Fortunately, there's an easy way to make a floor and ensure that, no matter how, it's scaled so that the top remains on the XY plane.

3. In **Edit** mode (*Tab*), with all points selected (which should be by default; *A* is the hotkey if it's not), move (*G*) them along the Z axis (*Z*) 1 unit (- + *1*):
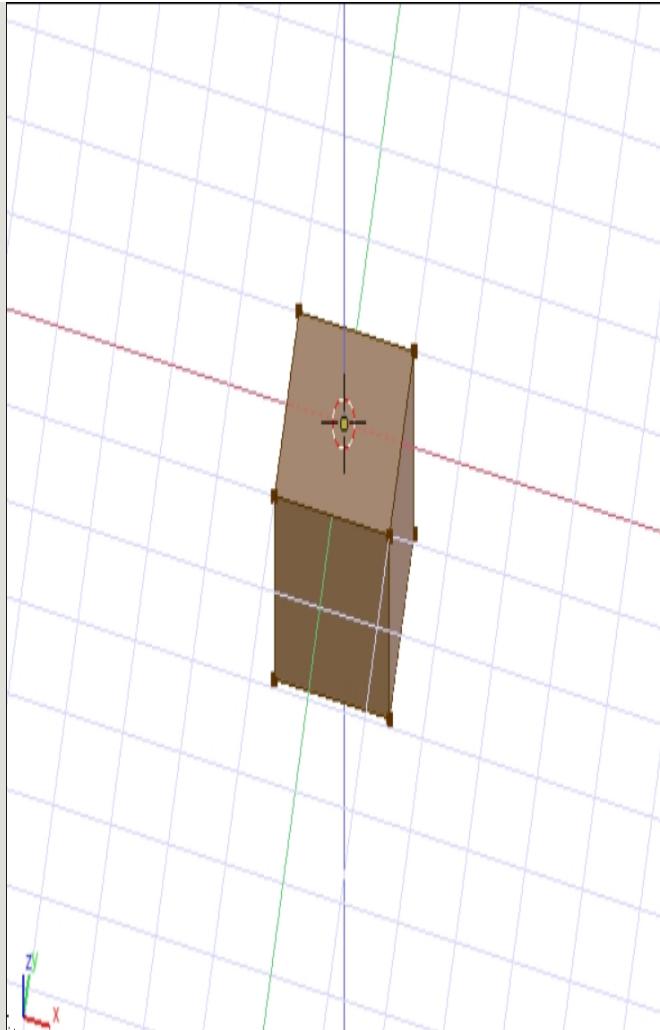
4. Then exit **Edit** mode (*Tab*) and scale (*S*) the cube. Notice that the top remains on the XY plane. This is because object transformations are made in relation to the object's origin. When the points were all moved in **Edit** mode, the origin wasn't affected. So now when scaling the top, because it's in line with the cube's origin, it remains on the same plane:
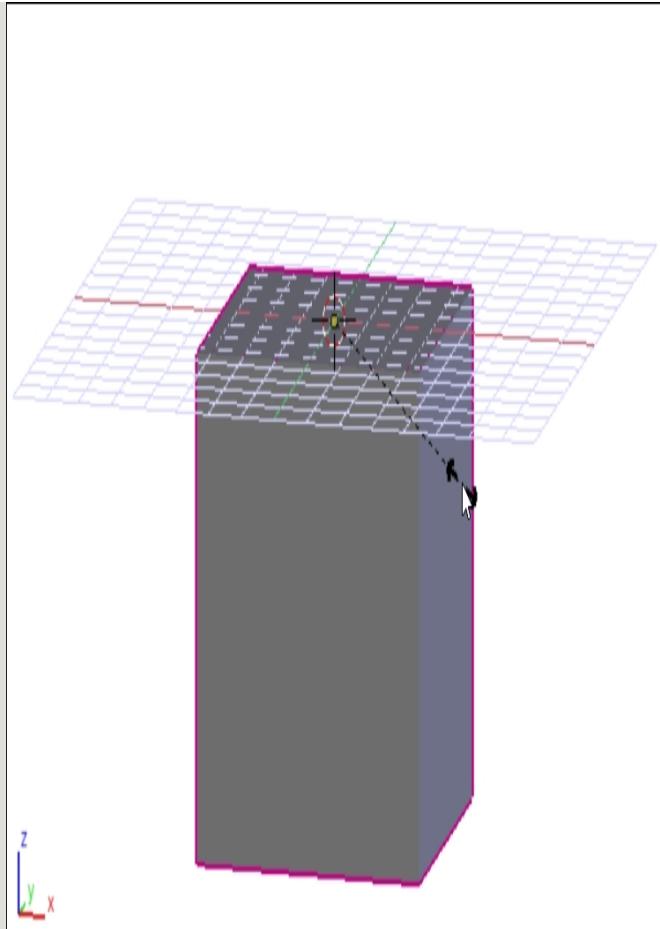
5. Exit **Local** view (Numpad slash /) and scale (*S*) the cube until it covers the bottom of the octopus's body:
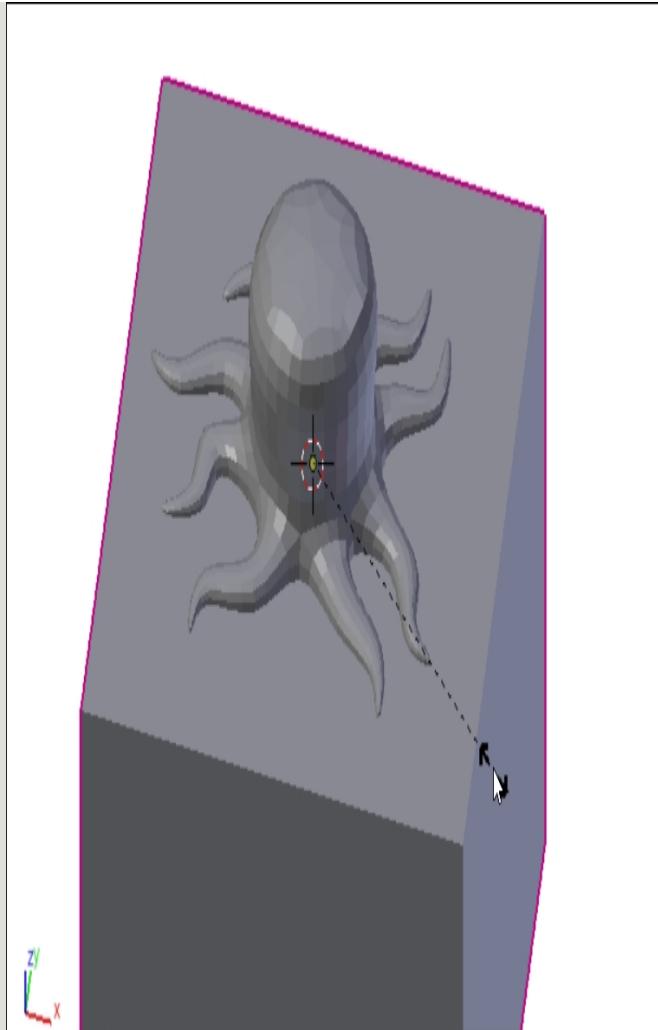
In **Solid** view (*Z*), it may look as if the octopus has a flat bottom, but in **Wireframe** view (*Z*), it is clear that the cube is only hiding the bottom part. For the final model, the bottom needs to be actually flat:
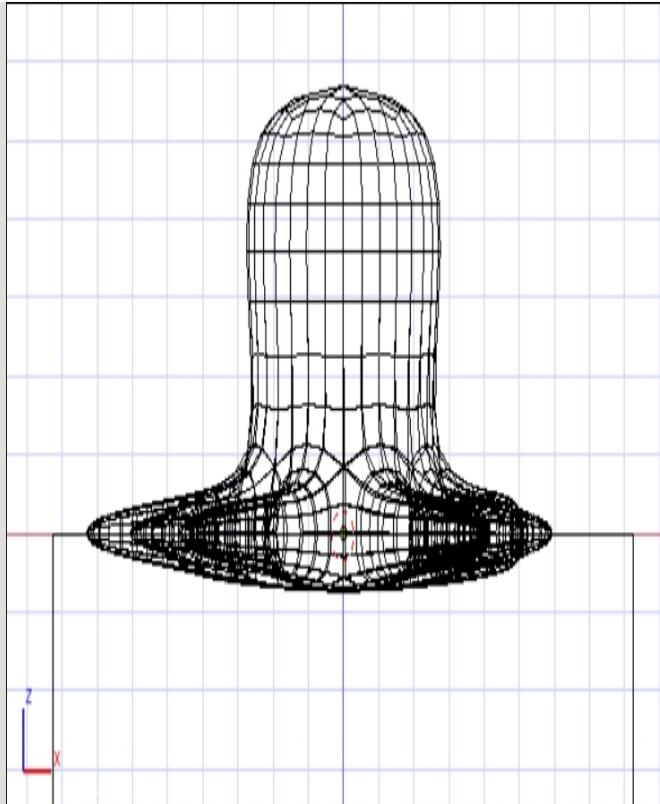
6. Select the octopus body.
7. In the **Modifier** tab, add a **Boolean** modifier to the octopus body:
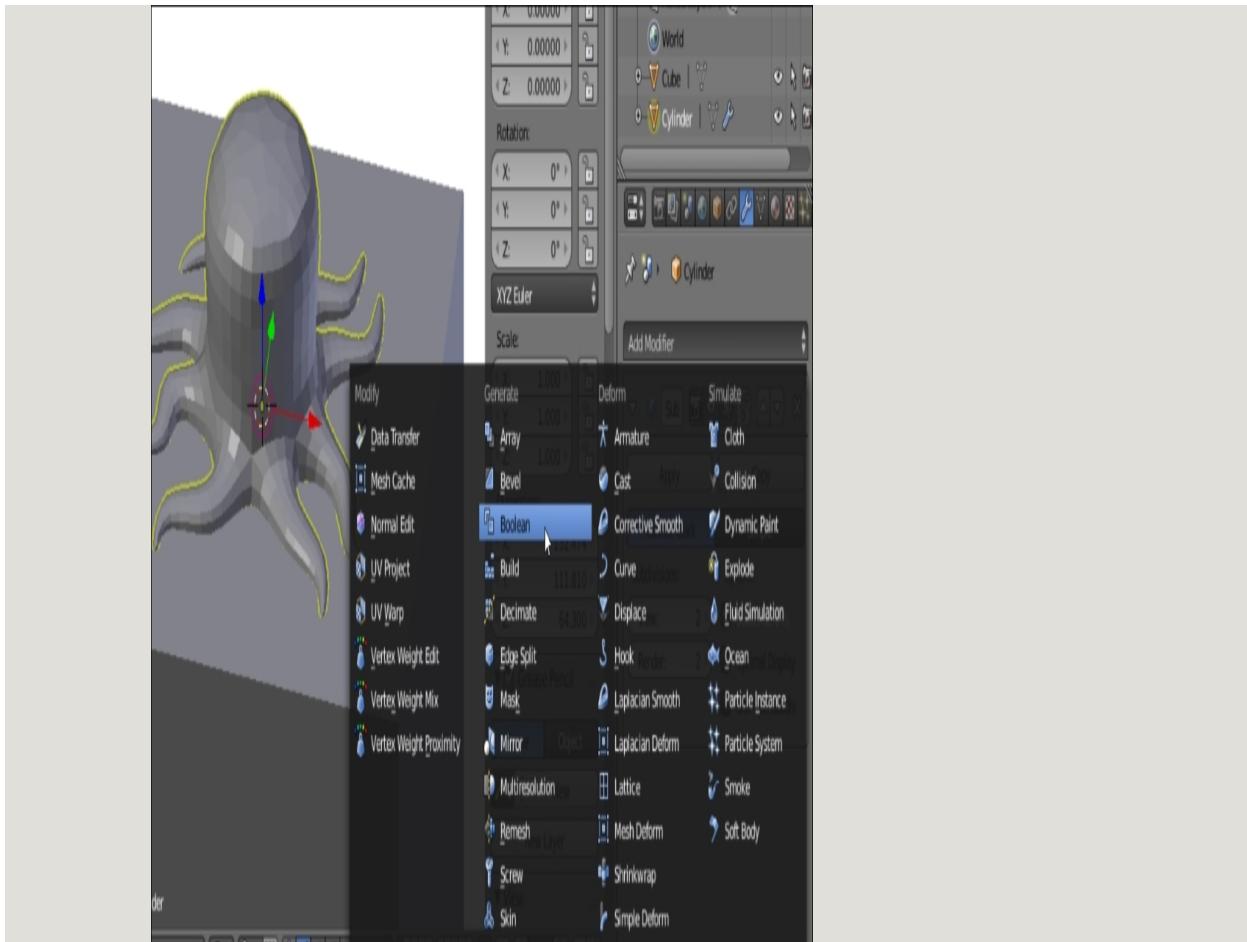
8. The **Boolean** modifier has options to combine the selected object with another object in a number of ways: **Intersect**, **Union**, and **Difference**. **Union** joins the two objects so that they become one, **Difference** cuts the second object out of the first, and **Intersect** leaves only the part where the two objects overlap. **Boolean** is a powerful tool and it's good to be aware of how to use it:
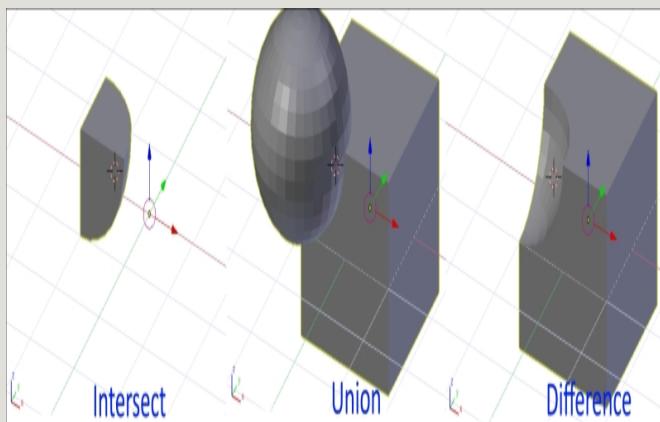
9. Change the **Operation** value of the **Boolean** modifier to **Difference**. Click on the **Object** text field and choose **Cube** from the list that pops up:
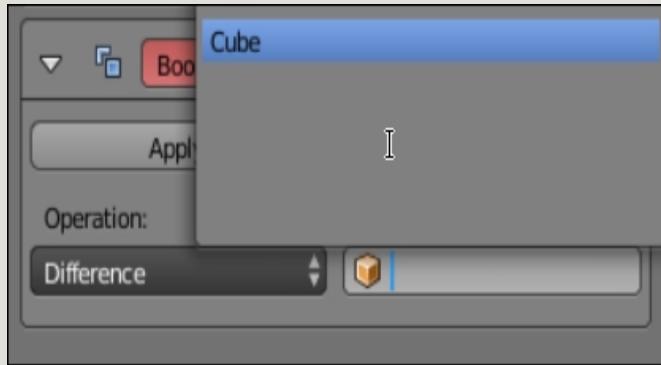
10. Now, the octopus body actually has a flat bottom suitable for 3D printing. This can be confirmed in **Wireframe** view (*Z*):
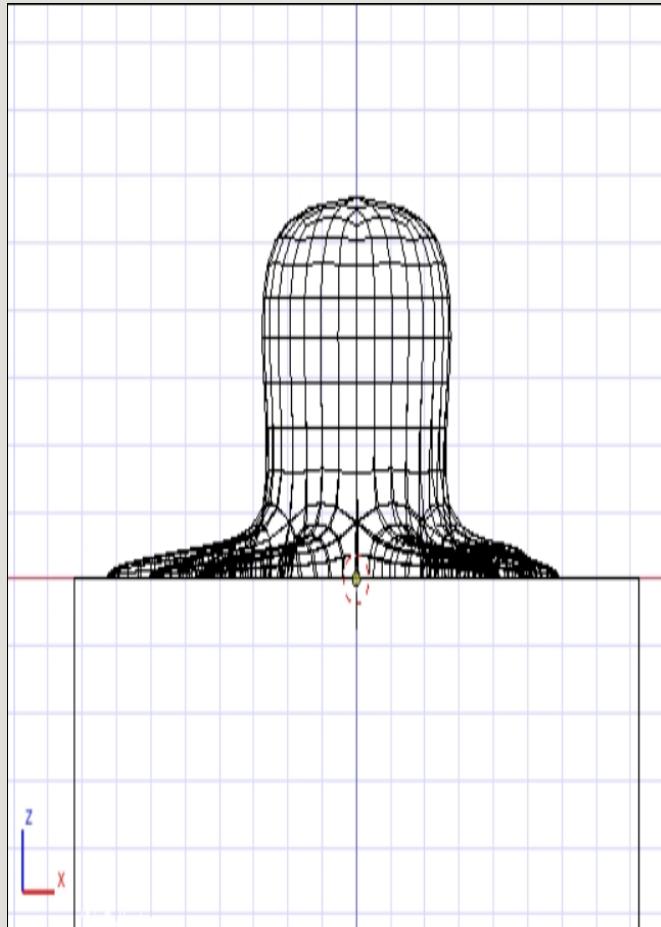
Renaming objects

Now that there is more than one object in the scene, leaving their names as the basic shapes they started as can be confusing, especially as other objects enter the scene. It's best practice to name objects as something more descriptive. Let's look at how to rename an object.

Objects can be renamed in the **Object** tab in the properties panel—the one marked with an orange cube icon:

1. Select the octopus body in the **3D View** (or **Cylinder** in the outliner panel). In the **Objects** menu, click on the name, currently **Cylinder**, and change it to **Octopus Body**:
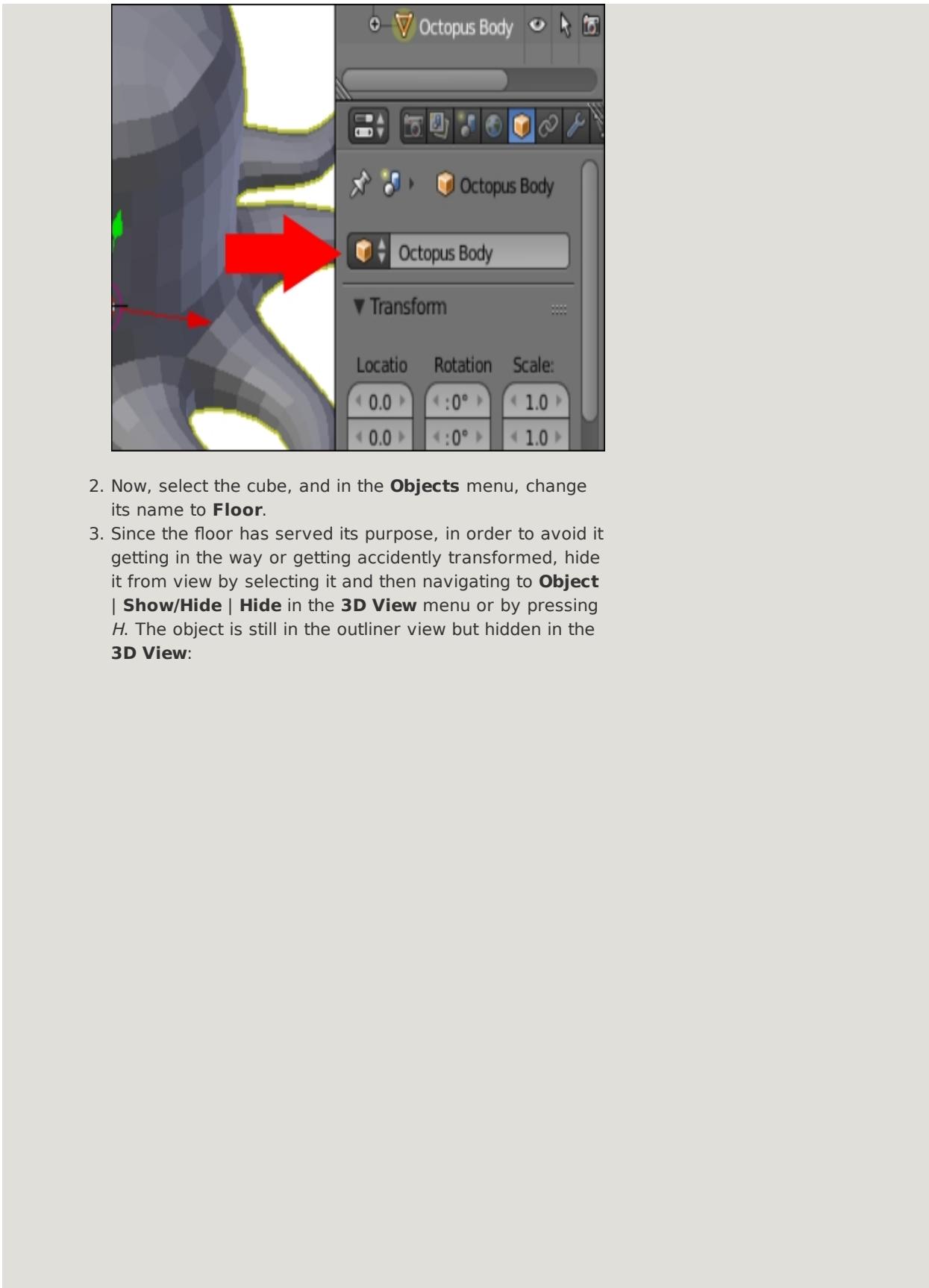
2. Now, select the cube, and in the **Objects** menu, change its name to **Floor**.

3. Since the floor has served its purpose, in order to avoid it getting in the way or getting accidently transformed, hide it from view by selecting it and then navigating to **Object | Show/Hide | Hide** in the **3D View** menu or by pressing *H*. The object is still in the outliner view but hidden in the **3D View**:
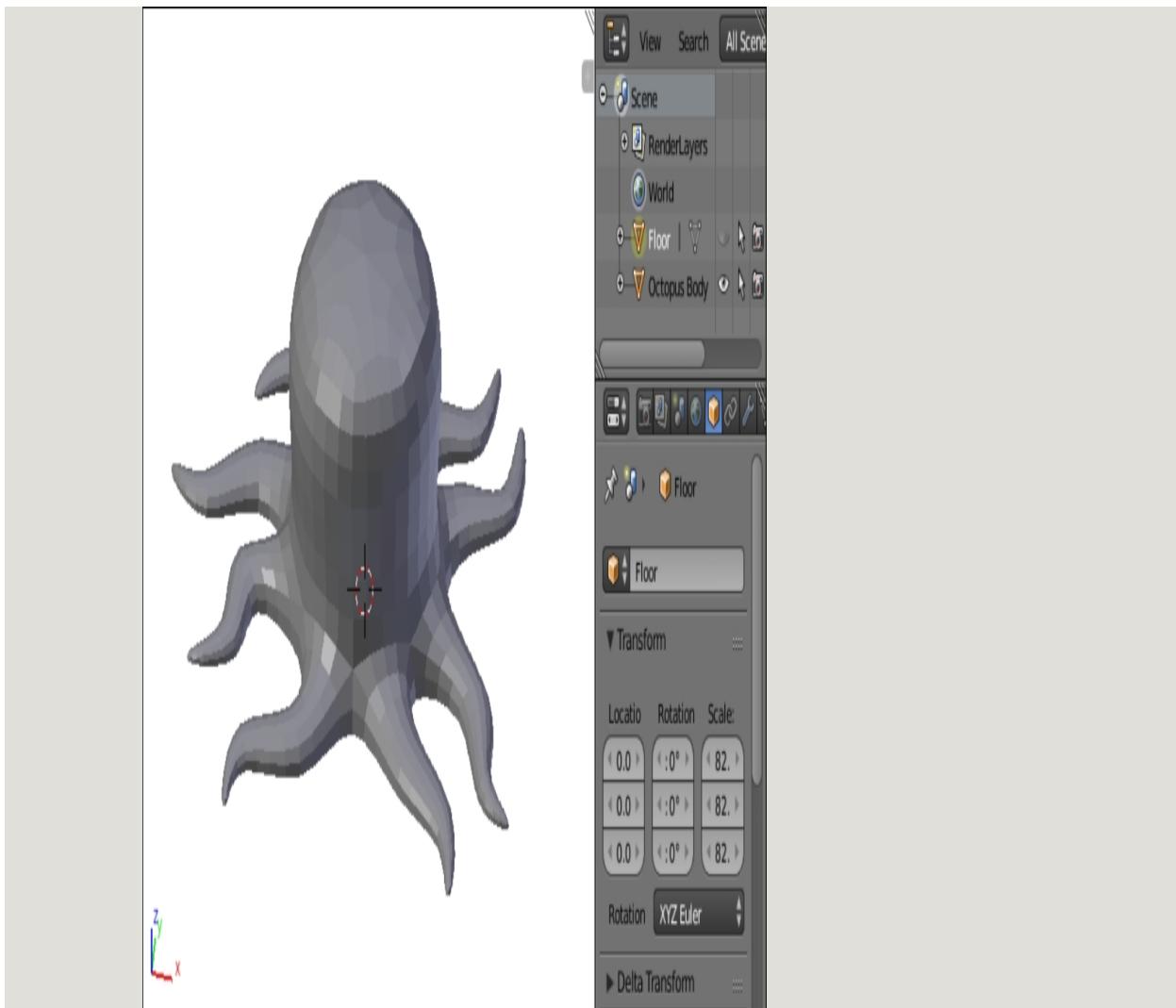
Hidden objects can be unhidden by going to
Object | **Show/Hide** | **Show All**, pressing *Alt +
H*, or by pressing the eye icon next to the object
in the **Outliner** panel.

Adding a pencil cup

The octopus model so far is appealing, but it can also be made functional. The plan for this project was a cup holder, so it needs to have the shape changed so that pencils can be put inside it.

1. To start, temporarily turn off the **Subdivision Surface** modifier by locating the **Subsurf** modifier in the **Modifiers** tab and clicking on the eye icon for it. Now the simplified geometry is easier to work with:
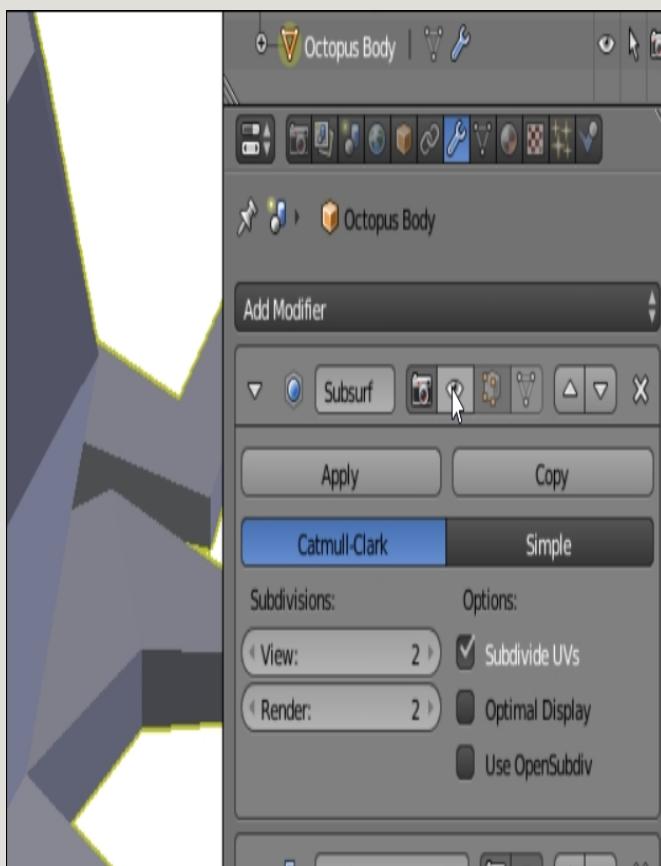
2. Go to **Face Select** (*Ctrl + Tab*) mode in **Edit** mode (*Tab*). Now, select the topmost face of the octopus:
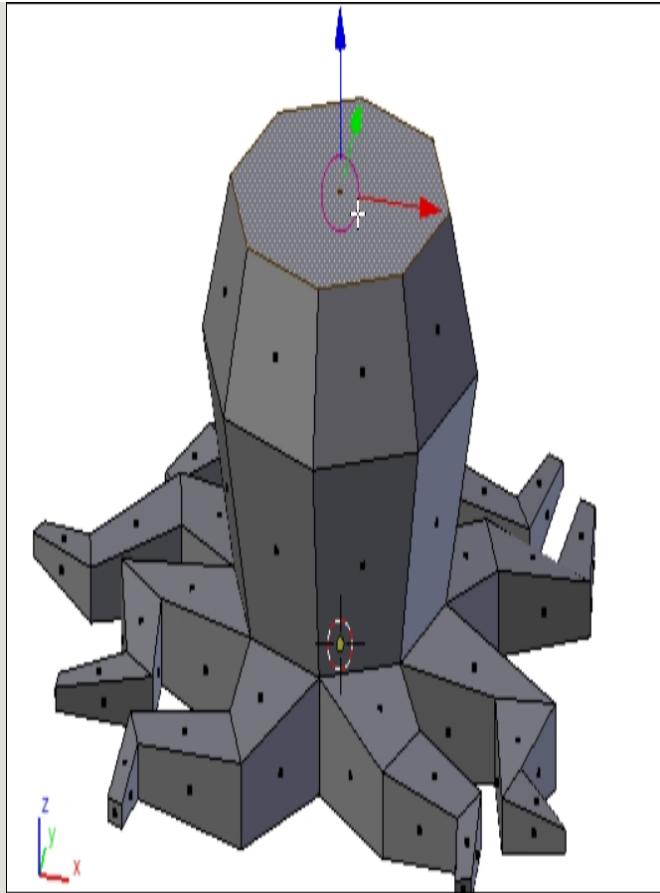
3. Switch to **Wireframe** (*Z*) and front **Ortho** view (Numpad *1*) and **Extrude** (*E*) the top face down into the body. Stop just a little above the red X axis line:
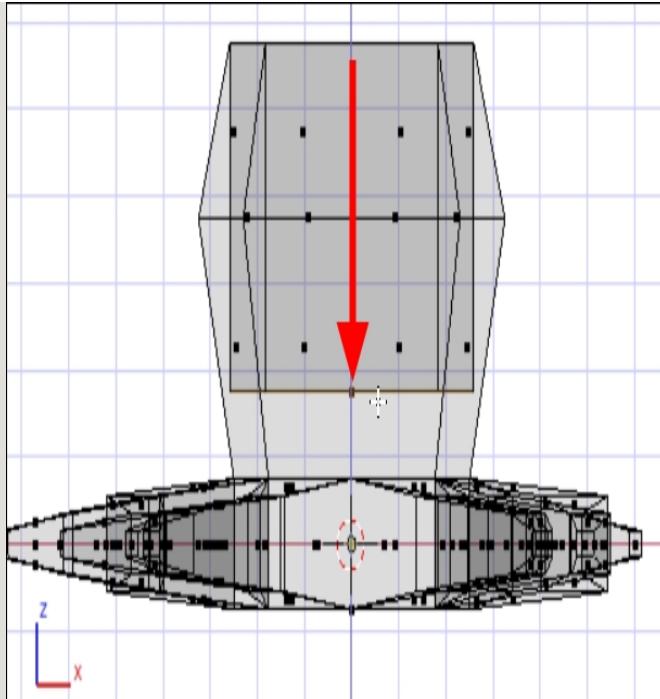
NOTE

It may be necessary to scale (*S*) the bottom of the cup a little so that it fits inside the body.
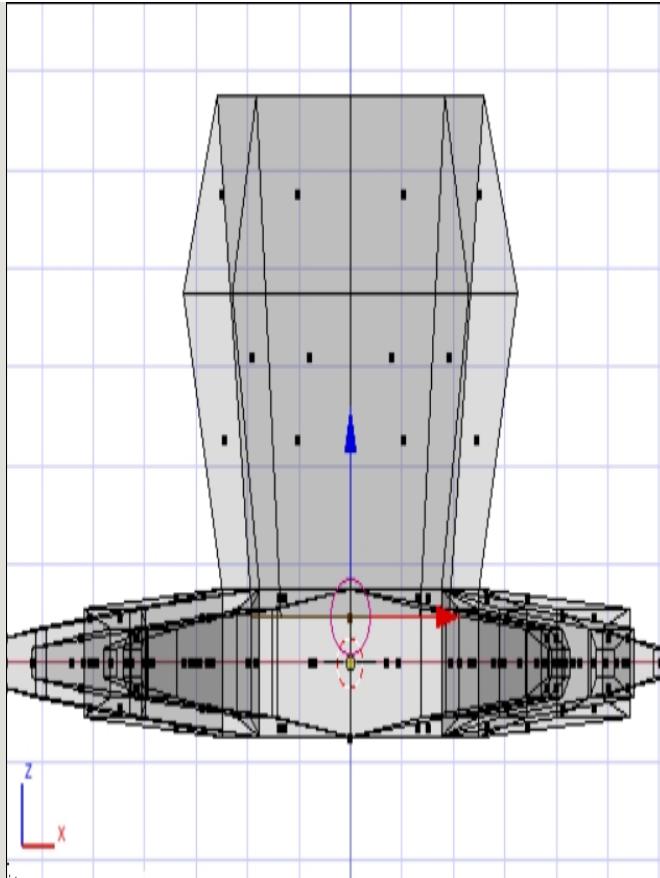
4. Now the shape is generally right for a pencil holder. Turn the **Subdivision Surface** modifier back on by clicking on its eye icon in the **Modifiers** tab. You'll notice that the cup bottom is a bit too round to be a space-efficient cup:
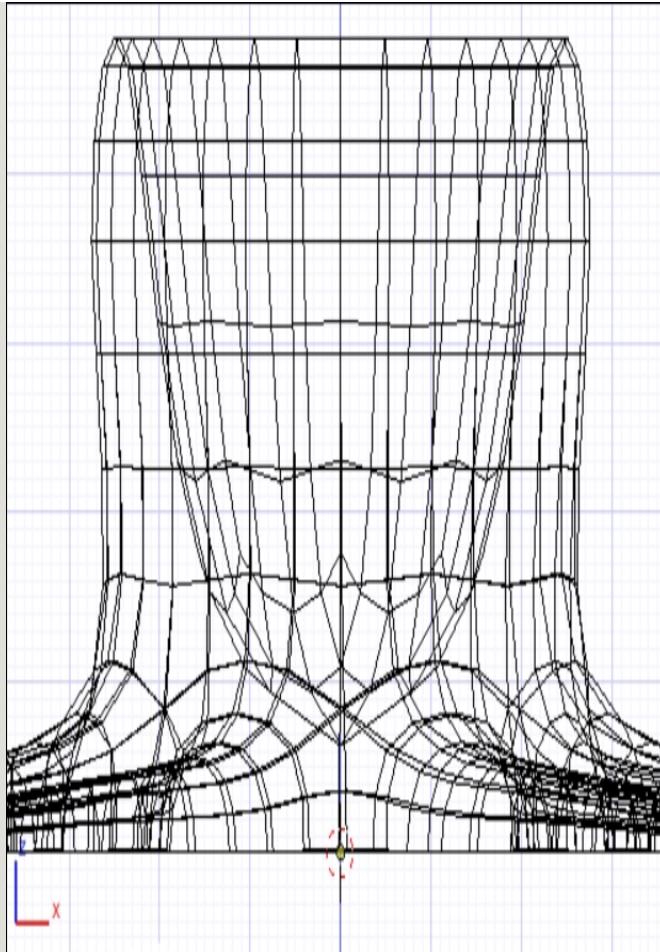
5. In **Edit** mode, edges can be marked with a "**crease**" to indicate to the **Subsurf** modifier that they should be sharpened.
6. With no points selected (*A*), in edge or point select mode (*Ctrl + Tab*), hold down *Alt* on the keyboard and select (click on) one of the edges around the top lip of the cup to select all the points in a loop around the top of the mug:
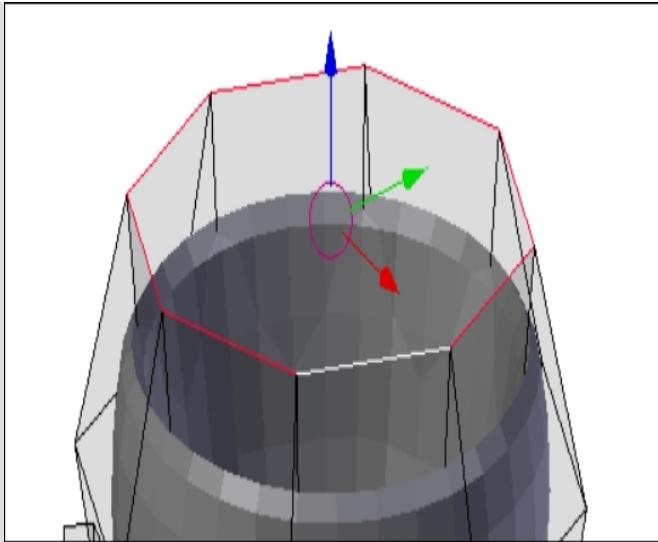
7. From the **3D View** menu, go to **Mesh** | **Edges** | **Edge Crease** or press *Ctrl + E* on the keyboard. Move the mouse pointer up or down to increase or decrease the amount of creasing applied to the edge, until it looks good:
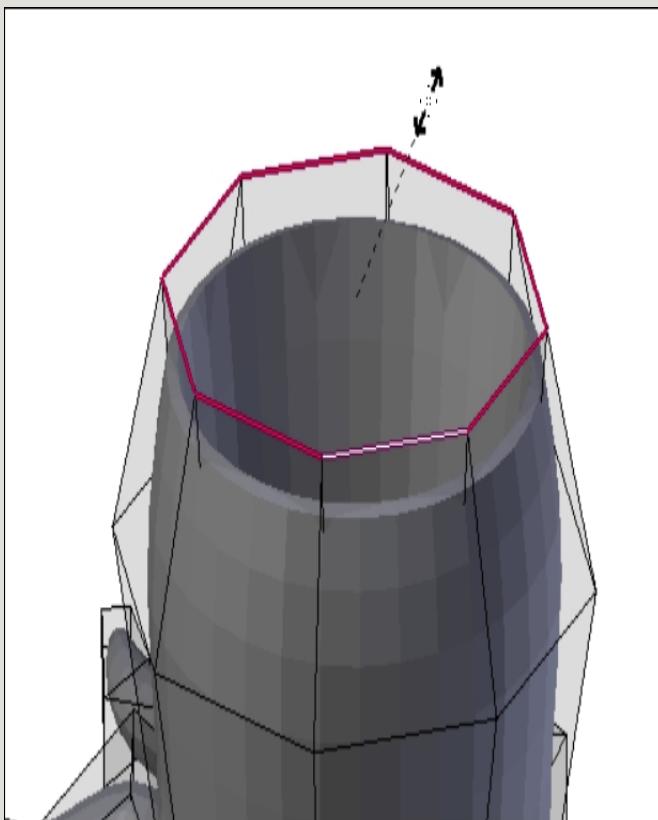
8. Hold *Alt* and select (click on) one of the edges at the bottom of the cup to loop select all the points around the

bottom of the cup. It may help to switch to **Wireframe** mode (*Z*) or temporarily turn off the **Subsurf** modifier:
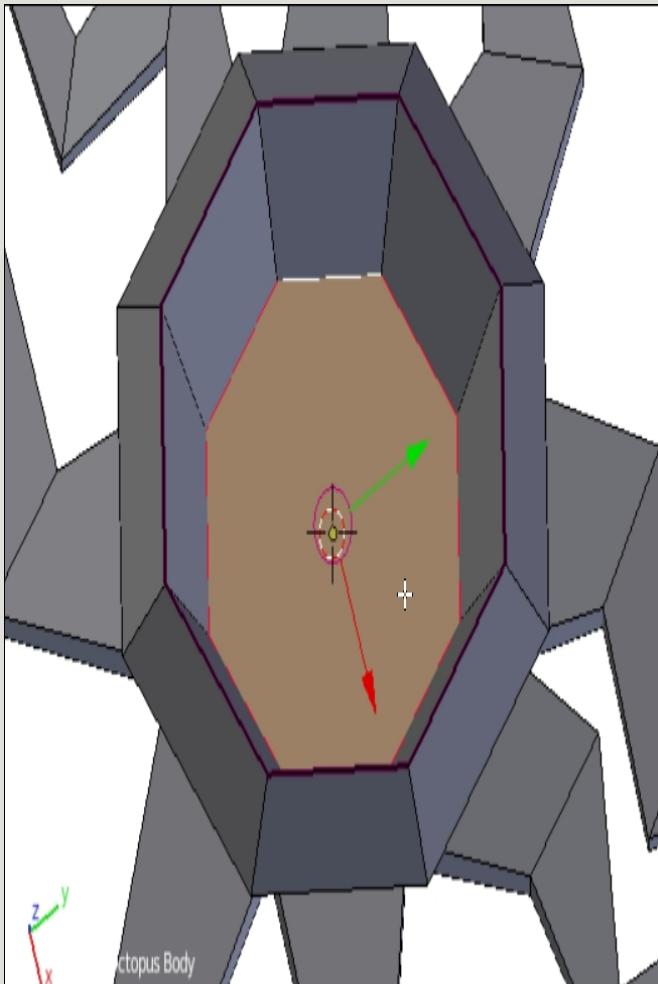
9. This time, set the crease value by finding the **Mean Crease** setting in the **Properties** tab (*N*). This setting can be any decimal number from 0.0 (off) and 1.0 (maximum). Click on the setting and enter 1, and then press *Enter* to set the value. Now the bottom of the cup is flat:
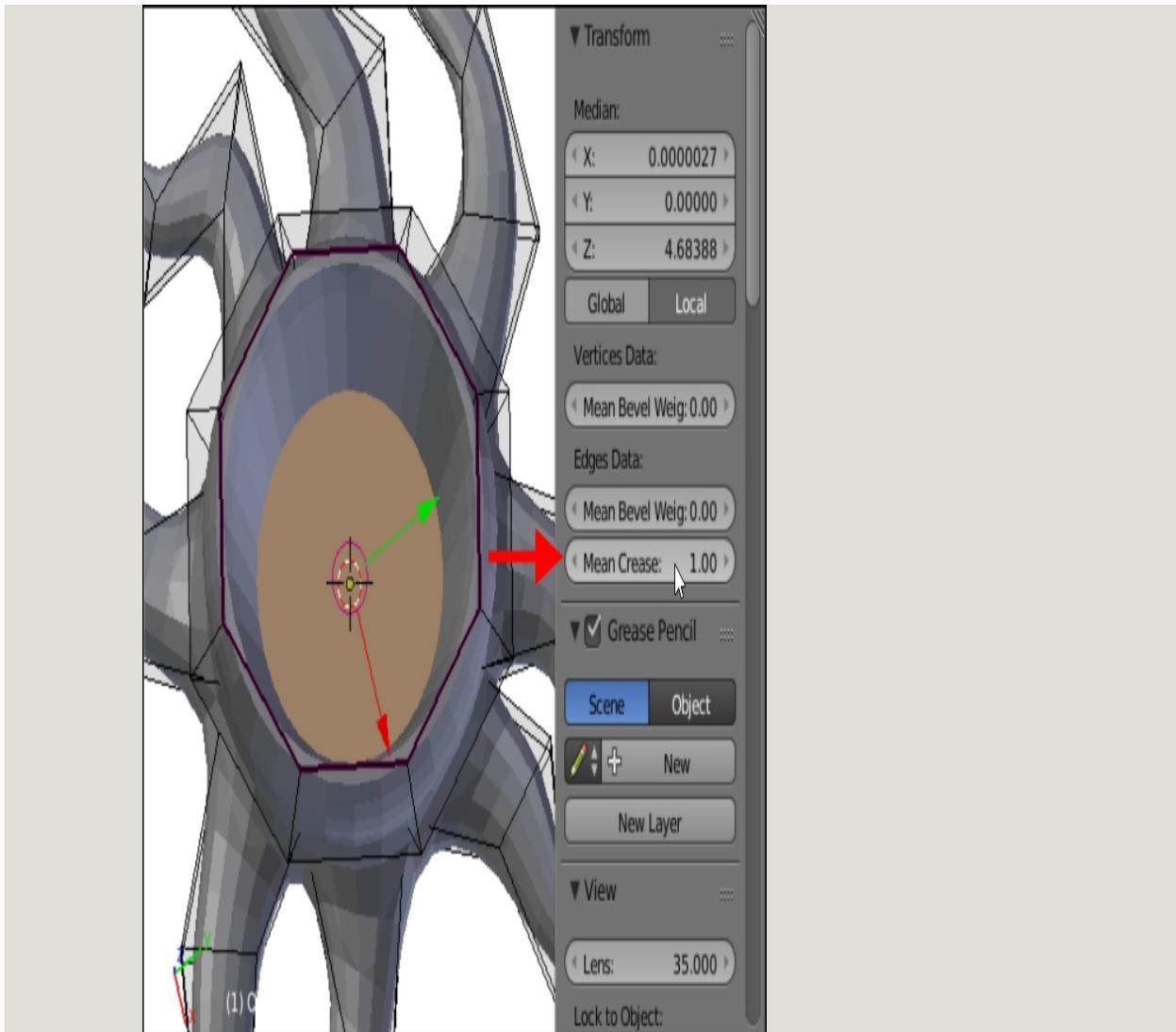
This sort of edge creasing is a powerful way to control the **Subsurf** modifier. Notice that the bottom of the cup is still circular even though the original mesh is an octagon. **Subsurf** still smooths out the other edges even if it's told to crease some—it's very smart!

Adding a face

Functionally, the design is complete, but as the old saying goes, it's all in the details. This little octopus would be much cuter with a face. Here's how you give it one:

1. Make sure **Edit** mode is off (*Tab*) and that the 3D cursor is at the origin (*Shift + C*).
2. Create a circle (*Shift + A*) by navigating to **Mesh | Circle**:
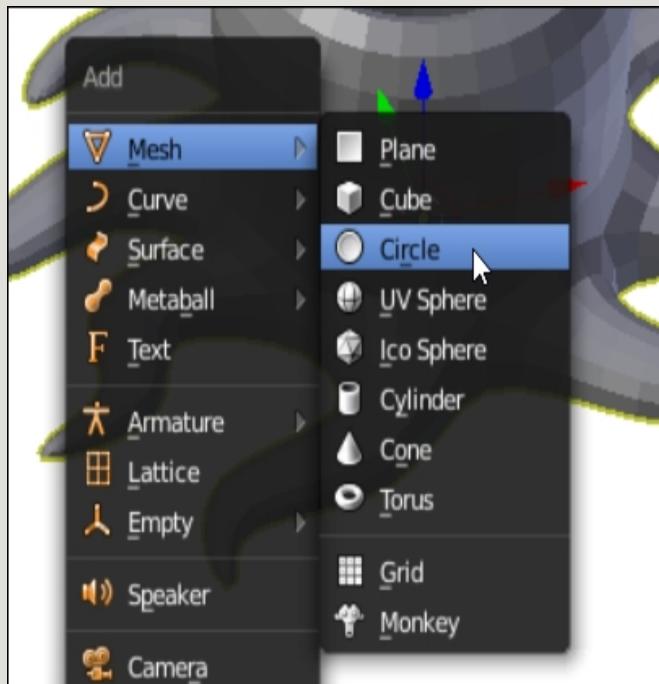
3. Again, the newly created circle is hidden inside the octopus, so with the circle still selected, switch to **Local** view (Numpad /).

   Since this circle will become the face, it's good to name it Face now, using the same steps for naming objects from before.
4. A circle in Blender is a flat object that is just a ring of points with no face. But that's easy enough to fix:
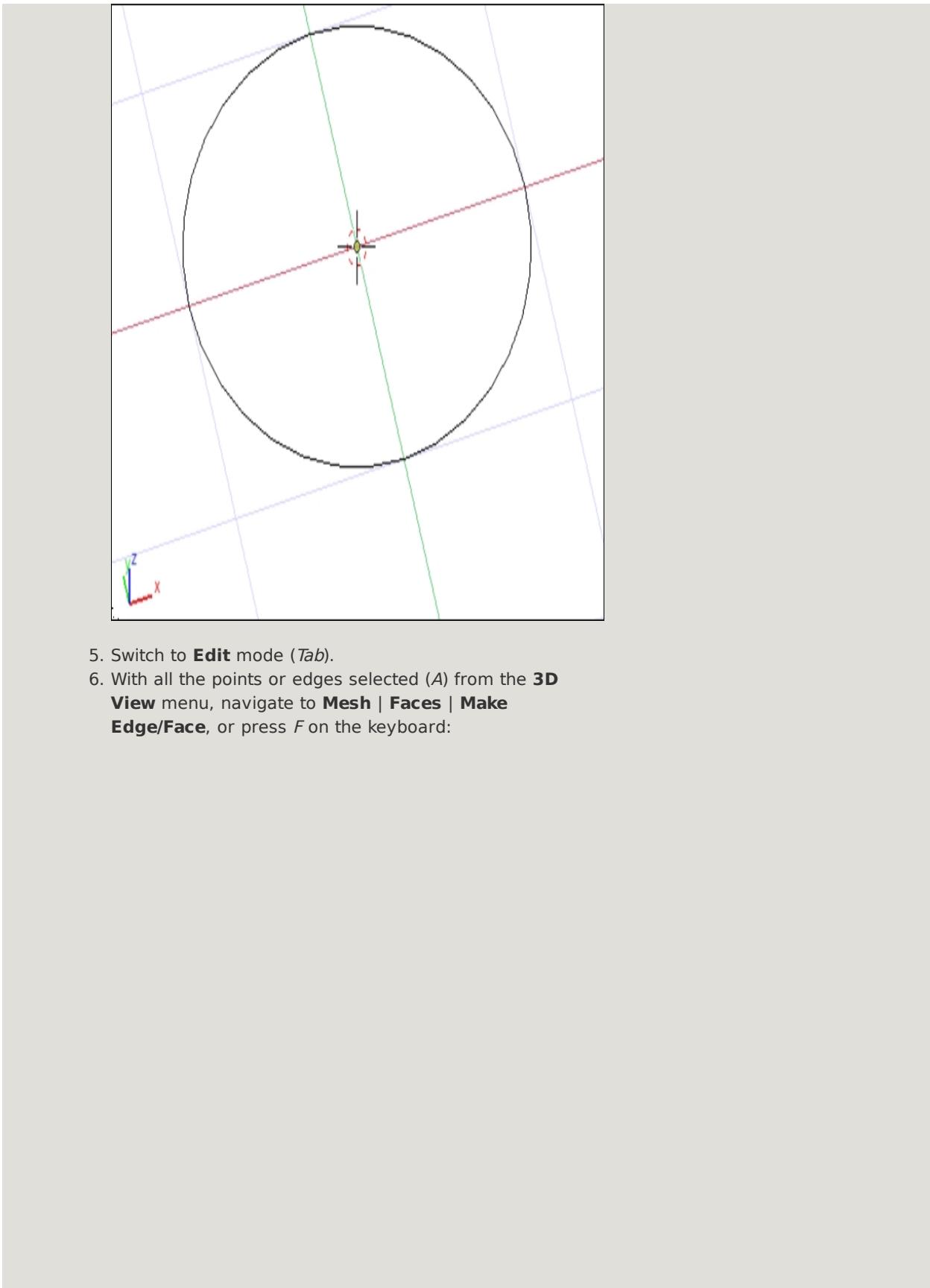
5. Switch to **Edit** mode (*Tab*).
6. With all the points or edges selected (*A*) from the **3D View** menu, navigate to **Mesh** | **Faces** | **Make Edge/Face**, or press *F* on the keyboard:
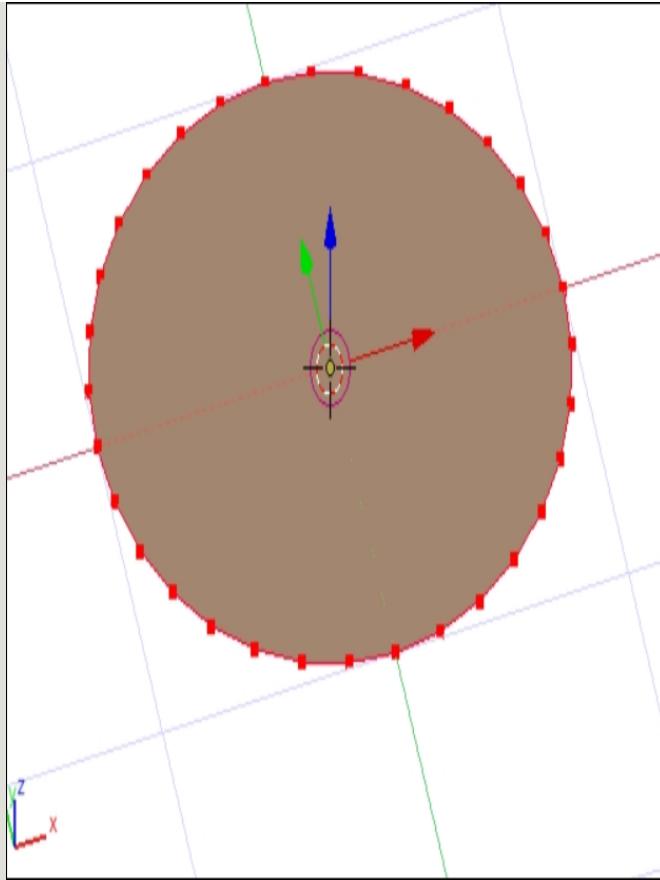
The Make Edge/Face command attempts to make a connected face from the selected vertices or lines. It can be quite intelligent, but if the points are not all flat, the edges may twist unexpectedly.

7. While still in **Edit** mode, switch to top view (Numpad *7*), and with all the points still selected, move (*G*) them to the left of the green Y-axis line:
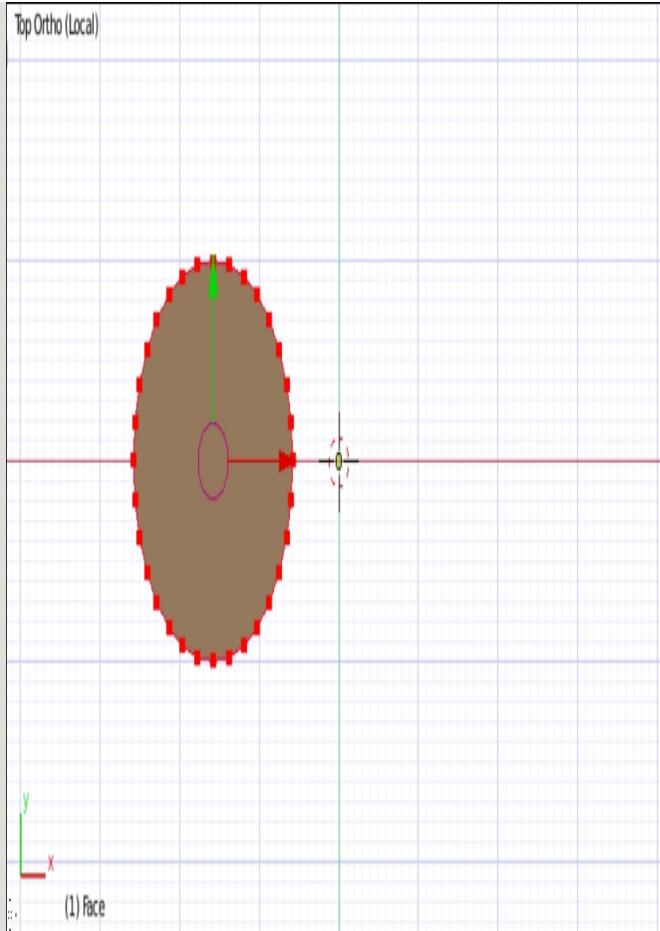
Duplicating in **Edit** mode doesn't create a new object. Whatever is selected is duplicated, but it remains a part of the object it was duplicated from. Points, lines, and faces can all be duplicated and transformed within the same object, even if it looks like they're separate parts. In this way, a face with separate eyes and mouth can be created that will still all be one face object.

8. Duplicate (*Shift + D*) all the points and place the duplicated points on the opposite side of the green Y axis line:

y

x

(1) Face
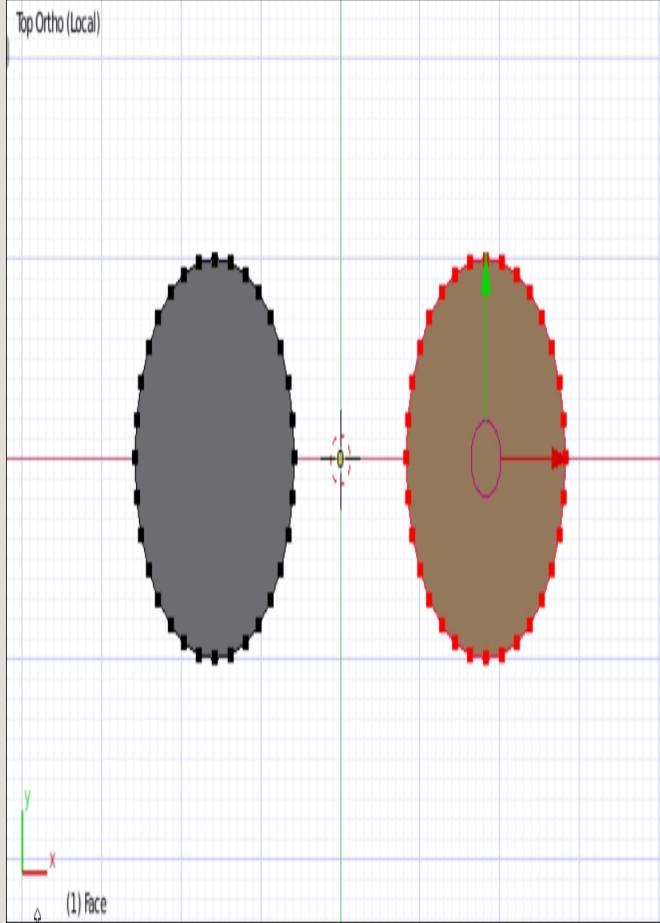
Both eyes are still a part of the face object, even though they're separate. In the same way, objects created while in **Edit** mode are likewise added to the existing object.

9. While still in **Edit** mode (making sure the 3D cursor is still at the origin), add (*Shift* + *A*) a plane to the face object by navigating to **Mesh** | **Plane**:

(1) Face

10. Move (*G*) the plane down along the Y-axis (*Y*) (about 3 units):

Top Ortho (Local)

(1) Face

11. Deselect all the points (*A*).
12. Select the two points at the top of the plane.
13. Scale (*S*) them up about twice to make the square a smiling mouth:

Top Ortho (Local)

(1) Face
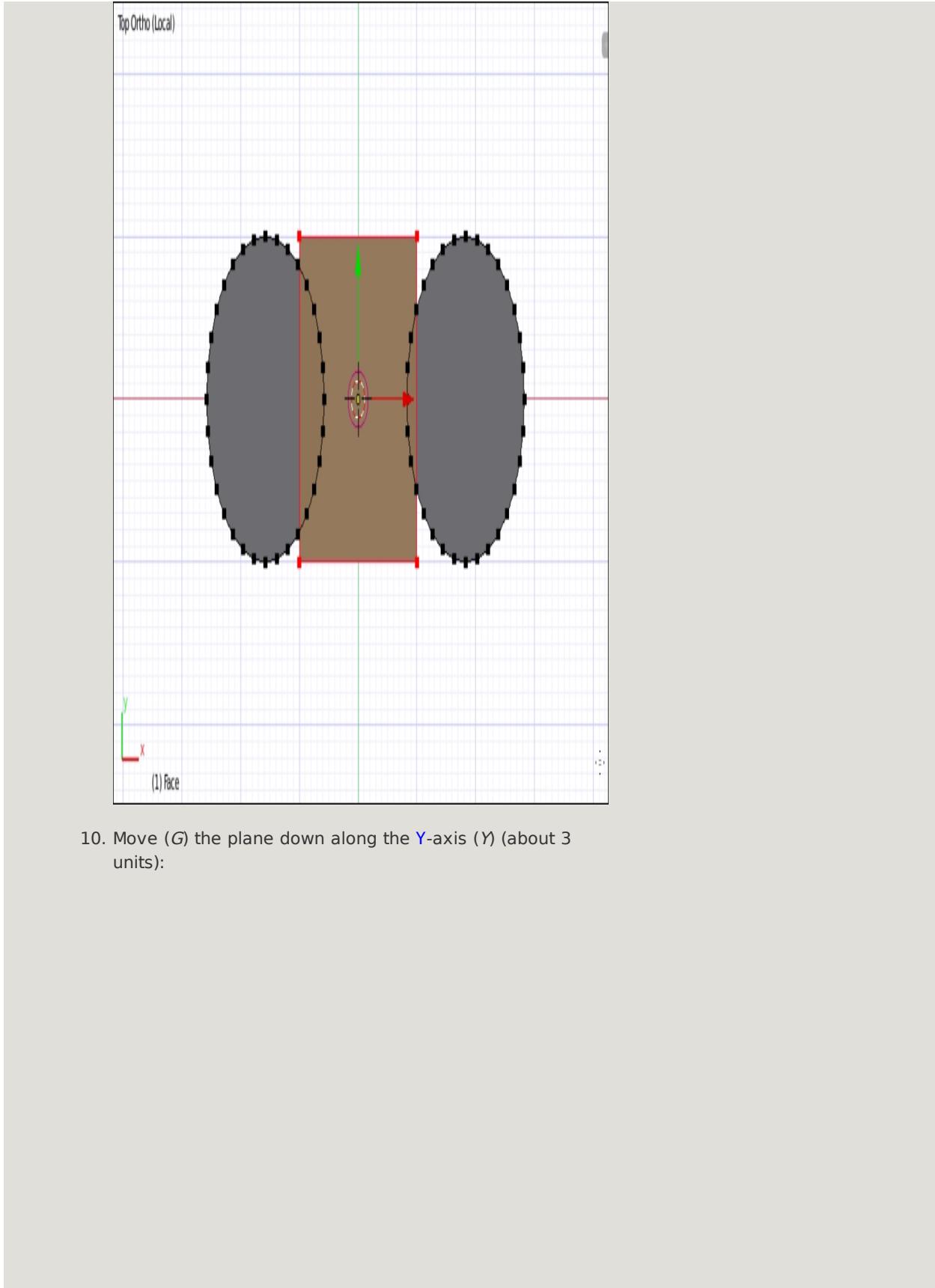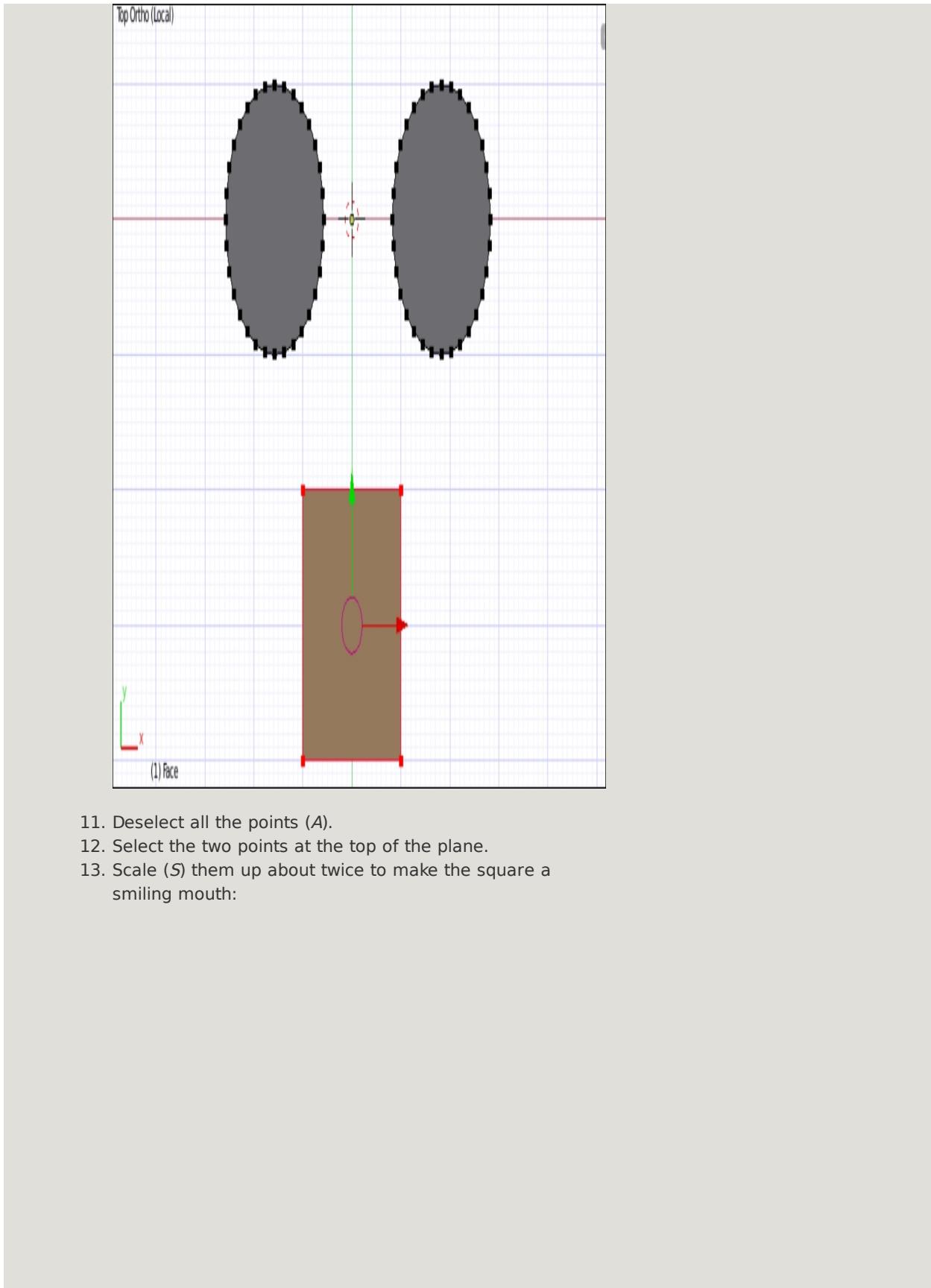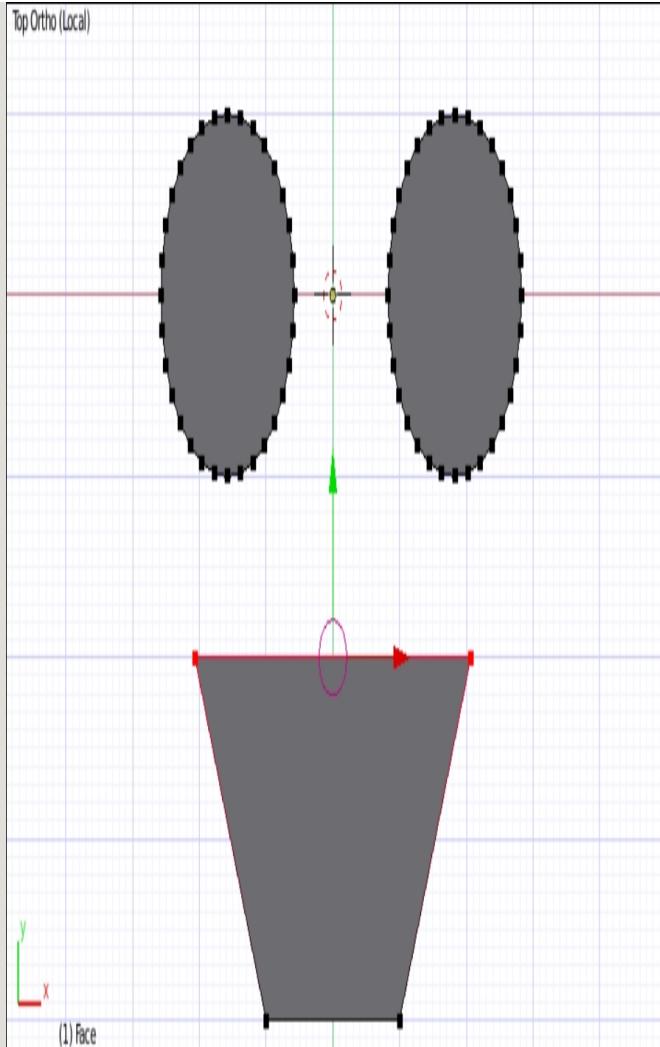
The shapes now look like a face, but only in two dimensions. If seen from a different angle, it's clear that the face is very flat. To give it the additional dimensions necessary, follow these steps:

14. Select all the points (*A* twice) and extrude them (*E*) about 4 or 5 units:
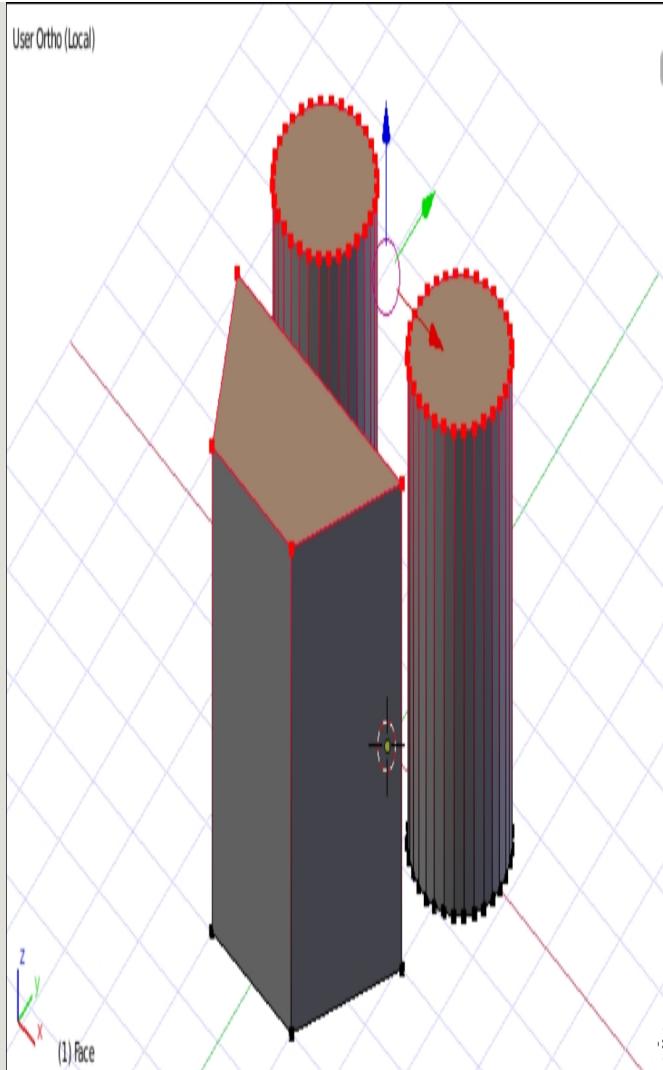
15. Exit **Edit** mode (*Tab*). The face template is now complete and just needs to be scaled and positioned.

16. Rotate (*R*) the face object around the X axis (*X*) 90 degrees (*90*):

User Ortho (Local)

(1) Face
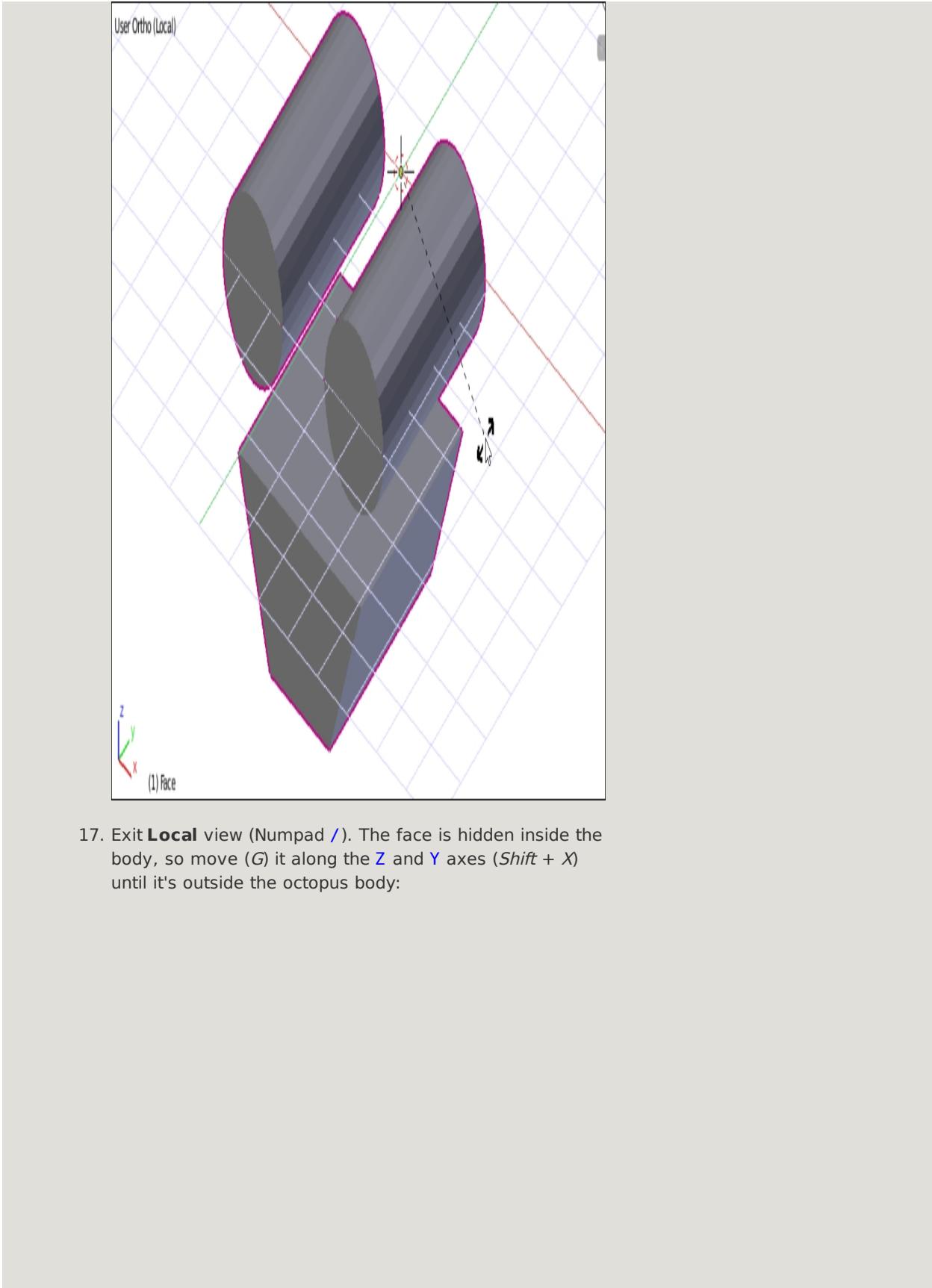
17. Exit **Local** view (Numpad /). The face is hidden inside the body, so move (G) it along the Z and Y axes (*Shift* + X) until it's outside the octopus body:

18. In front **Ortho** view (Numpad *1*), scale the face (*S*) and move (*G*) it along the Z axis (*Z*) until it's positioned properly:
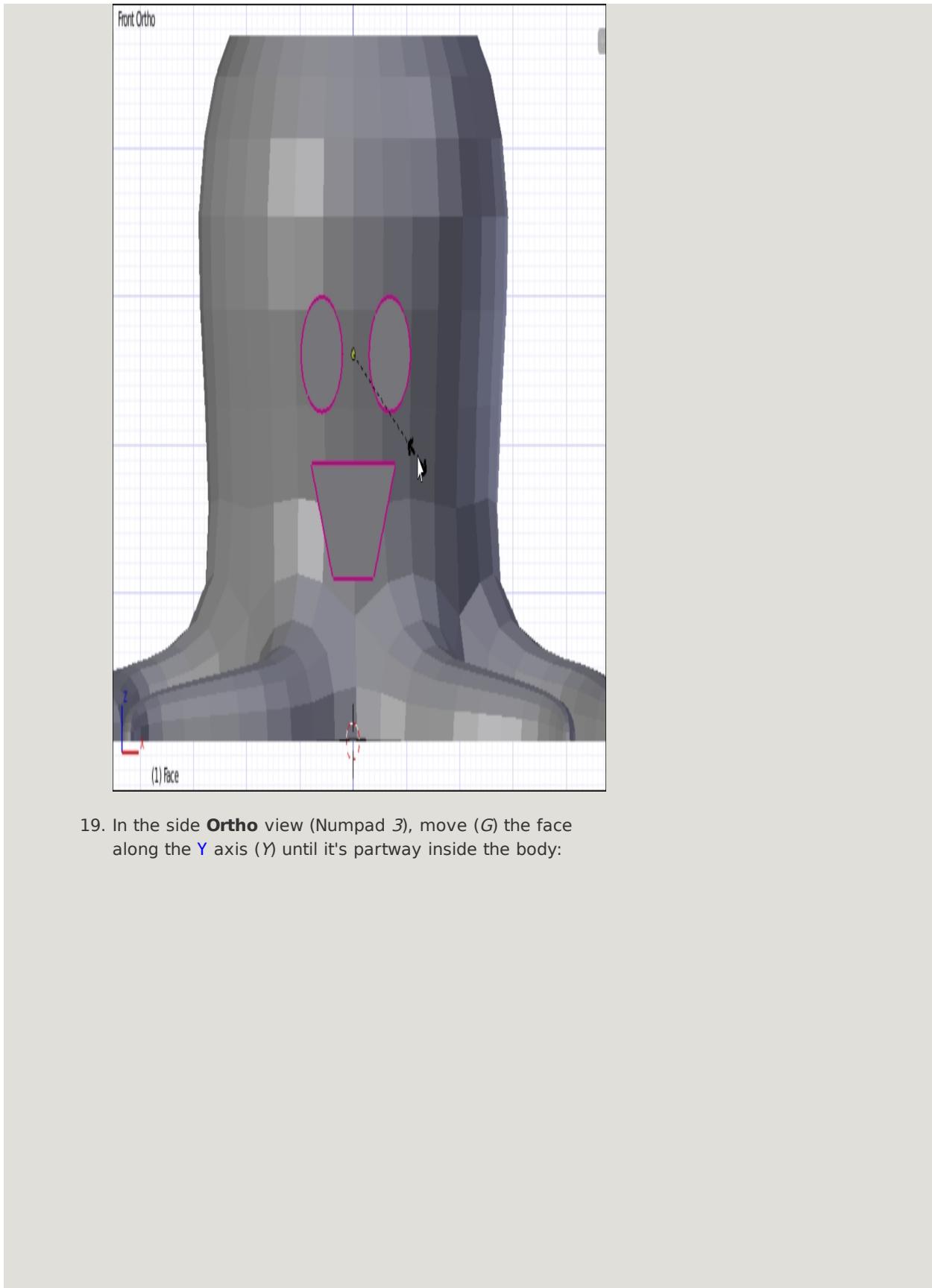
19. In the side **Ortho** view (Numpad *3*), move (*G*) the face along the **Y** axis (*Y*) until it's partway inside the body:
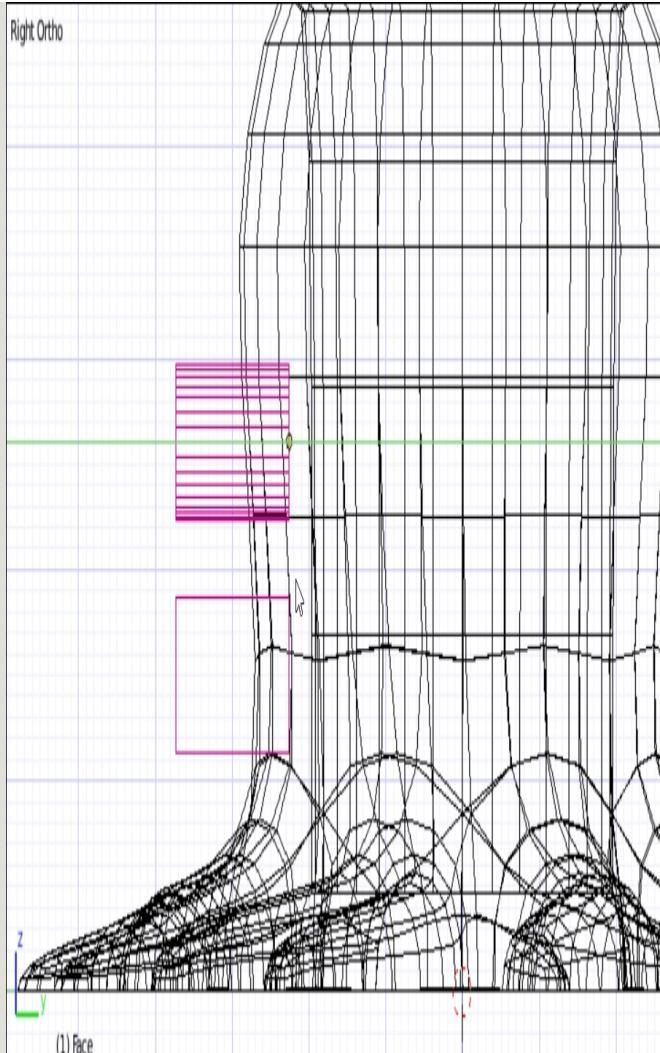
It may help to use **Wireframe** view to ensure the face is deep enough but not too deep. About four or five small squares' depth (or `0.4` or `0.5` real-world millimeters) is perfect. If it's too deep, it may cut all the way through into the cup space and create an overhang problem. The top of the mouth will rely on some bridging during prints, and that's okay.

20. Now, select the body and in the **Modifier** tab, add a **Boolean** modifier. Change the settings of the **Boolean** modifier to be a **Difference** operation with the face object:

| Difference | ▼ | 🟧 Floor | ⊠ |

| ▼ | 🔳 | Boolean.001 | 📷 👁 | △ ▽ | ✕ |

| Apply | Copy |

Operation: Object:

| Difference | ⇕ | 🟧 Face | ⊠ |
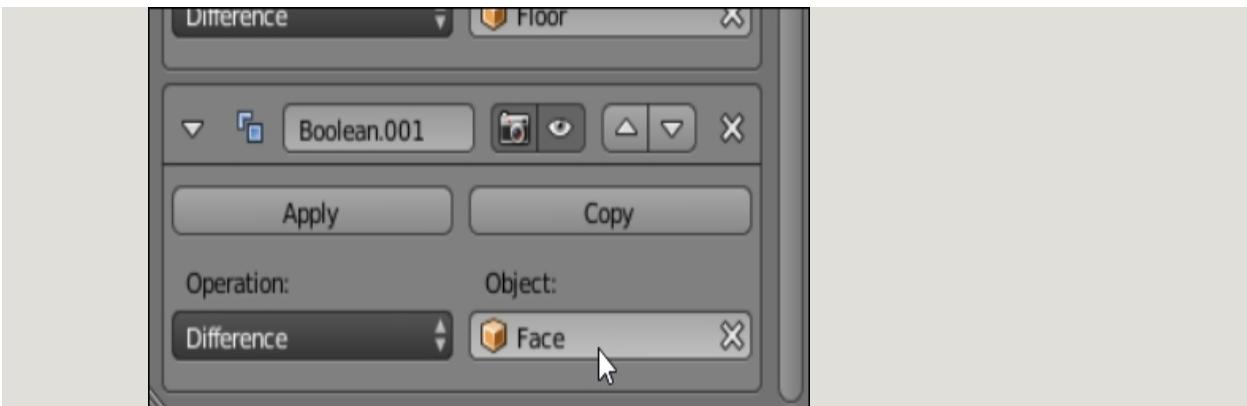
21. The effect of this new modifier won't be visible immediately, so select the face object and hide it (*H*):

User Ortho

(1) Octopus Body
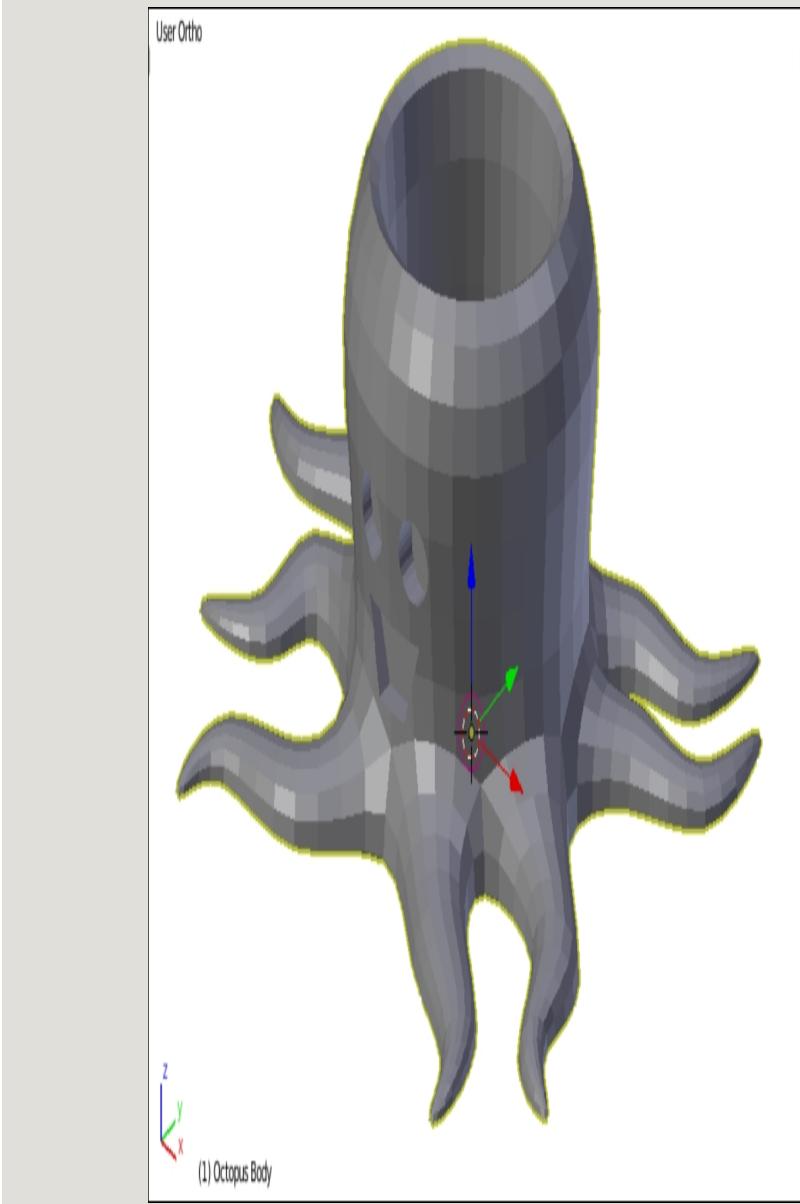
Your octopus pencil holder is now complete.

Select the body, and in the **Info Panel** menu (at the top of the screen), go to **File** | **Export** | **Stl (.stl)** to export the 3D print-ready octopus holder. Be sure to choose a directory where you can find the file later in order to send it to the printer.

Finishing touches

The octopus is done now, but don't stop there. Use your imagination to make this design your own. Add accessories and personality, change the base shape, and come up with your own design to make this something you'll want to share. Just remember the rules about overhangs in your design, and the sky is the limit:
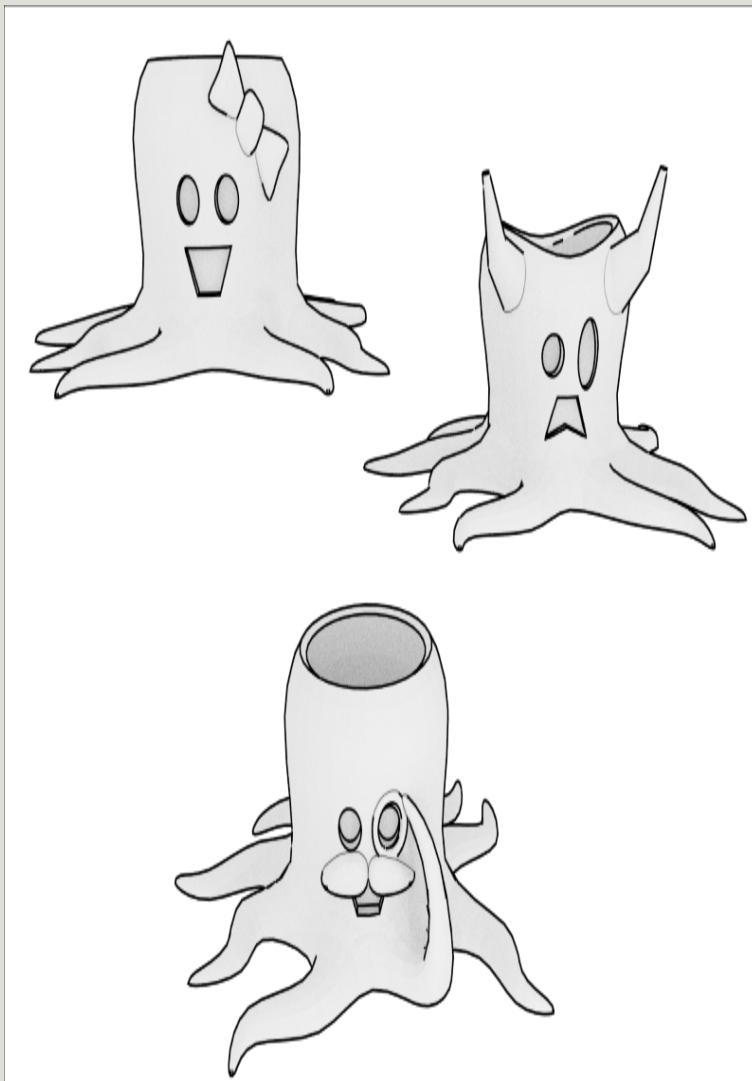
Summary

This project was functional but also cute. The techniques for vertex editing basic shapes in **Edit** mode with extrusion and loop cuts and then using the mesh smooth and **Boolean** modifiers to further modify them can be used to create an endless number of projects of any type and shape. Not all designs need to be cute, but they also don't all need to be purely practical. Combine a practical design with an aesthetic element to add personality and really take advantage of what 3D printing can offer.

Happy designing!

Chapter 4. Measuring Basics

3D printing makes virtual things real. Sometimes those things need to match back up with an object in the real world after they are made, such as a lid for a can or a joint between two poles. Sometimes you're printing a replacement for a broken part. So it is often very important that, when planning, accurate measurements are made and applied to the modeling of a 3D object.

Some measuring tools you may already have around your house. Some may require you to buy special tools you don't have but should probably consider getting.

In this chapter, we will cover:

- Measuring with a ruler
- Measuring with calipers
- Grid paper trace method
- 3D scanning

Measuring with a ruler

Most homes have a ruler on hand that can, in many cases, provide relatively accurate measurements. For irregularly shaped objects, it may be difficult to use a ruler. But if the object is flat on at least two adjacent sides, rulers work fine in a pinch:
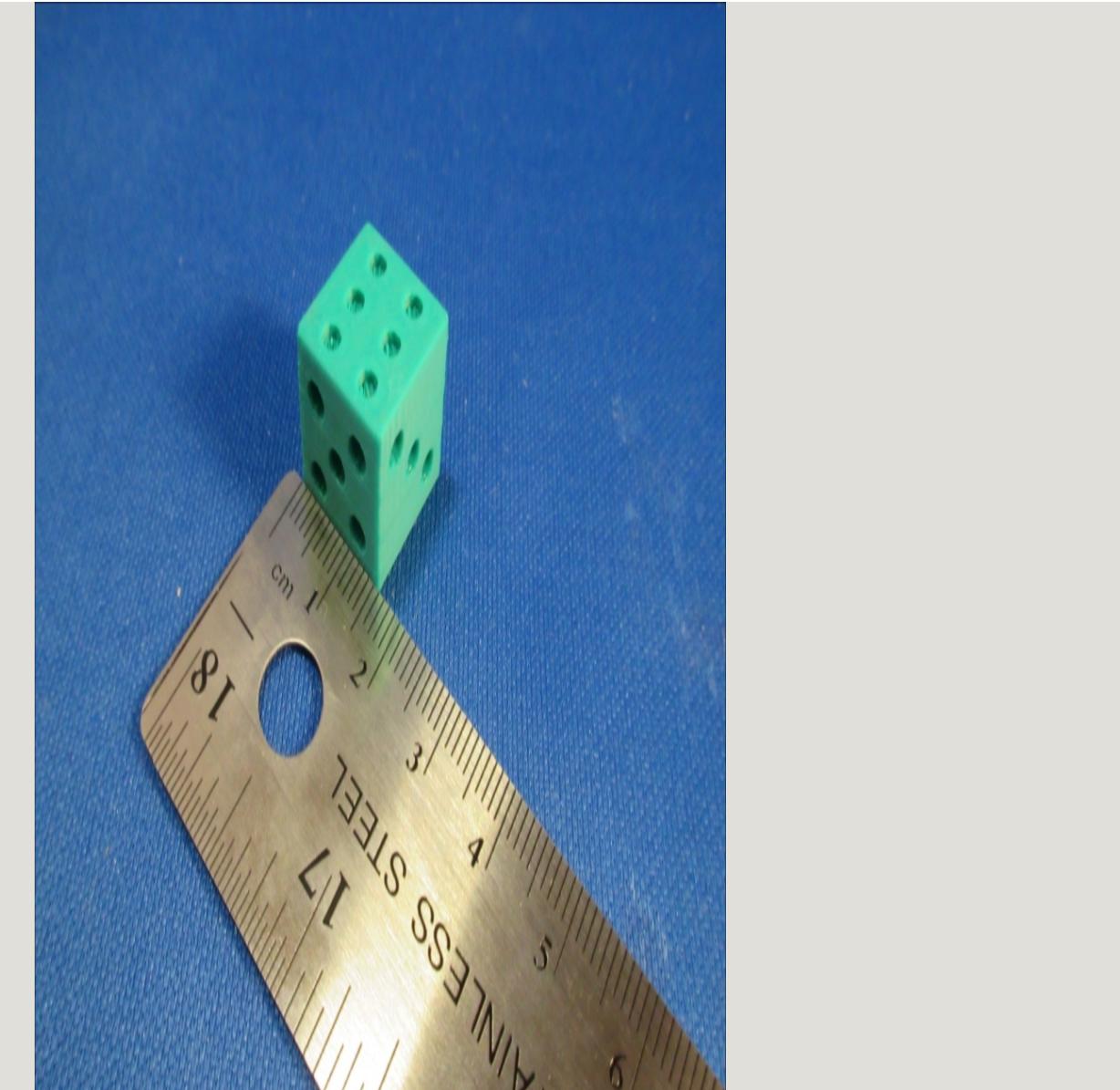
Simply lay the object on its flat edge, move one side to line up with the 0 mark, and measure the other side of the object. It's not the best method for measuring complex shapes, but sometimes a ruler can be enough.

Measuring with calipers

The most common technique for taking accurate measurements is the use of a tool called a **caliper**, a must-have for anyone who models for 3D printing. Calipers measure distance with a high degree of precision and can measure in three different ways: the outside diameter of an object with the outside jaws, the internal diameter with the inside jaws, or the depth with the depth probe at the far end. There are two common calipers: manual (or Vernier) calipers and digital calipers.

Manual or Vernier calipers

If budget is a concern, then perhaps a Vernier caliper is preferable, since they are generally less expensive than their digital counterparts. These calipers operate purely mechanically, but have a clever trick that allows them to be just as accurate if read properly:

Open the jaws to take a measurement and tighten them over the object to be measured. Then take a close look at the little window:
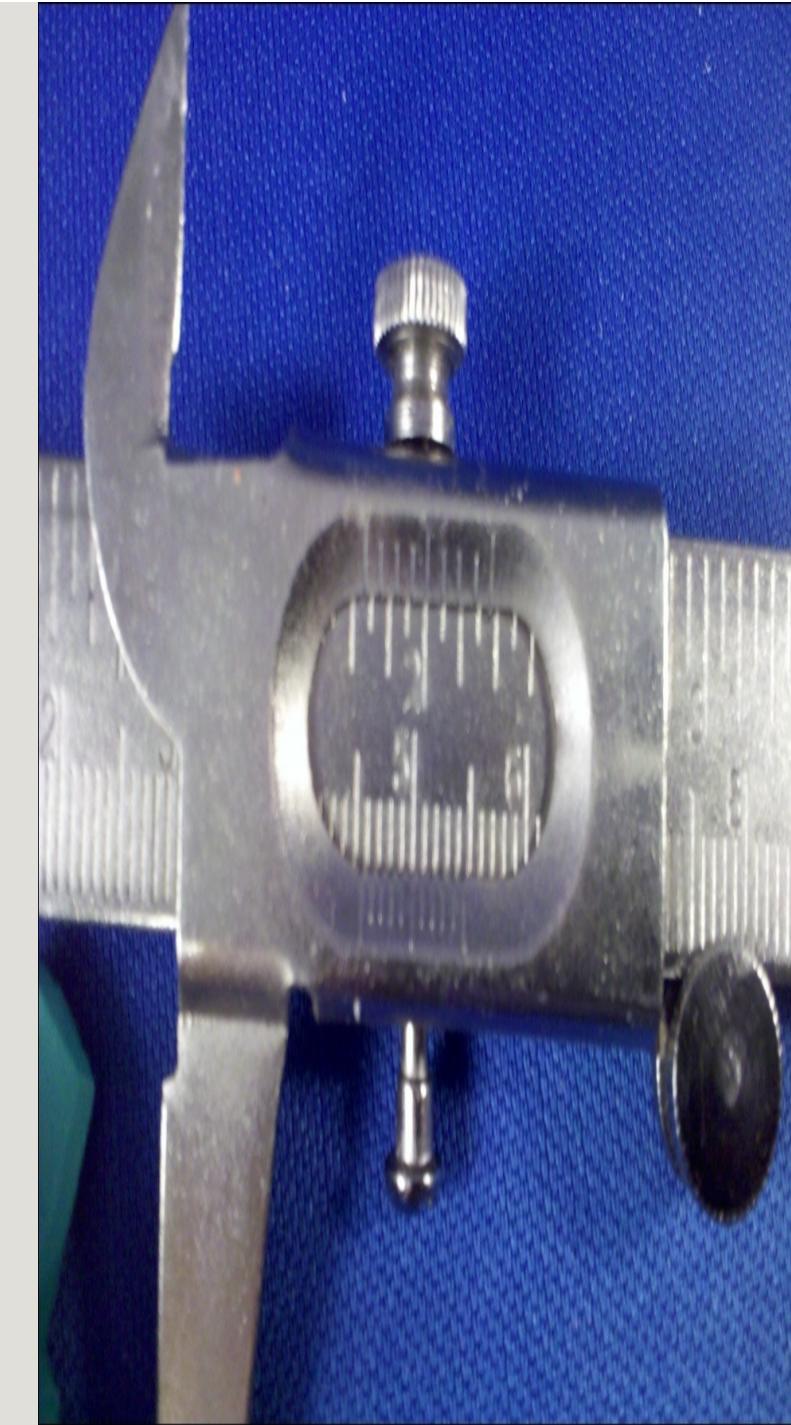
The leftmost tick is between 4.5 and 4.6, so this object is between 4.5 cm and 4.6 cm (or 45/46 mm). Then, count the tick marks on the outside until one of them lines up perfectly with one of the ticks on the inside. In this case the fifth tick mark on the outside lines up with a tick mark on

the inside. This is the hundredths part of the measurement, so this object measures 4.55 cm or 45.5 mm.

Vernier calipers never need to be calibrated. They don't need batteries, so they'll always work, and they're cheaper. But they do take some additional effort to read properly and lack the "cool" factor of their digital cousins.

Digital calipers

The easiest measuring tool to use is the digital caliper. Simply turn it on, tare or zero the reading while closed, then open the jaws, put the object to be measured between them, clamp it down, and take the reading. It's fast and accurate without much effort. Some models even have a port that can transfer the measurements directly to the computer:

But digital calipers have the disadvantage of being more expensive and relying on batteries. They can also go off calibration, reducing their accuracy. However, for quick and easy use, a digital caliper is hard to beat.

Whatever calipers you use, familiarize yourself with all parts for measuring different things. The large jaws on the front are for measuring the outside of objects. The small jaws opposite them are for measuring the inside of objects. At the end of the ruler section is a depth sensor that can be used to measuring the depth of something.

Grid paper trace method

There is another trick to measuring complex shapes that involves a common household item. An object with a complex shape can be traced on a piece of grid paper. That grid paper trace can then be scanned or photographed, uploaded to the computer, and imported into modeling software to recreate its shape. By tracing the shape on graph paper, there is a scale reference in the modeling software.

Suitable objects

The ideal object for this method of measuring is an object with a complex and otherwise difficult to measure shape but with at least one flat side so that it's easy to lay flat on a piece of paper. If the object doesn't have a flat side, being able to make one side flat will do the trick.

For example, this piece of plastic attaches to the back of a drawer to guide it along a rail so that it moves smoothly and straight:

The back of this piece is almost flat, except for 2 nubs sticking out, which would be easy enough to remove. Simply note the location of parts that are removed before removing them, since they'll need to be modeled back in.

Object preparation

To begin, use a sharp blade and cut the nubs off the mostly flat surface:
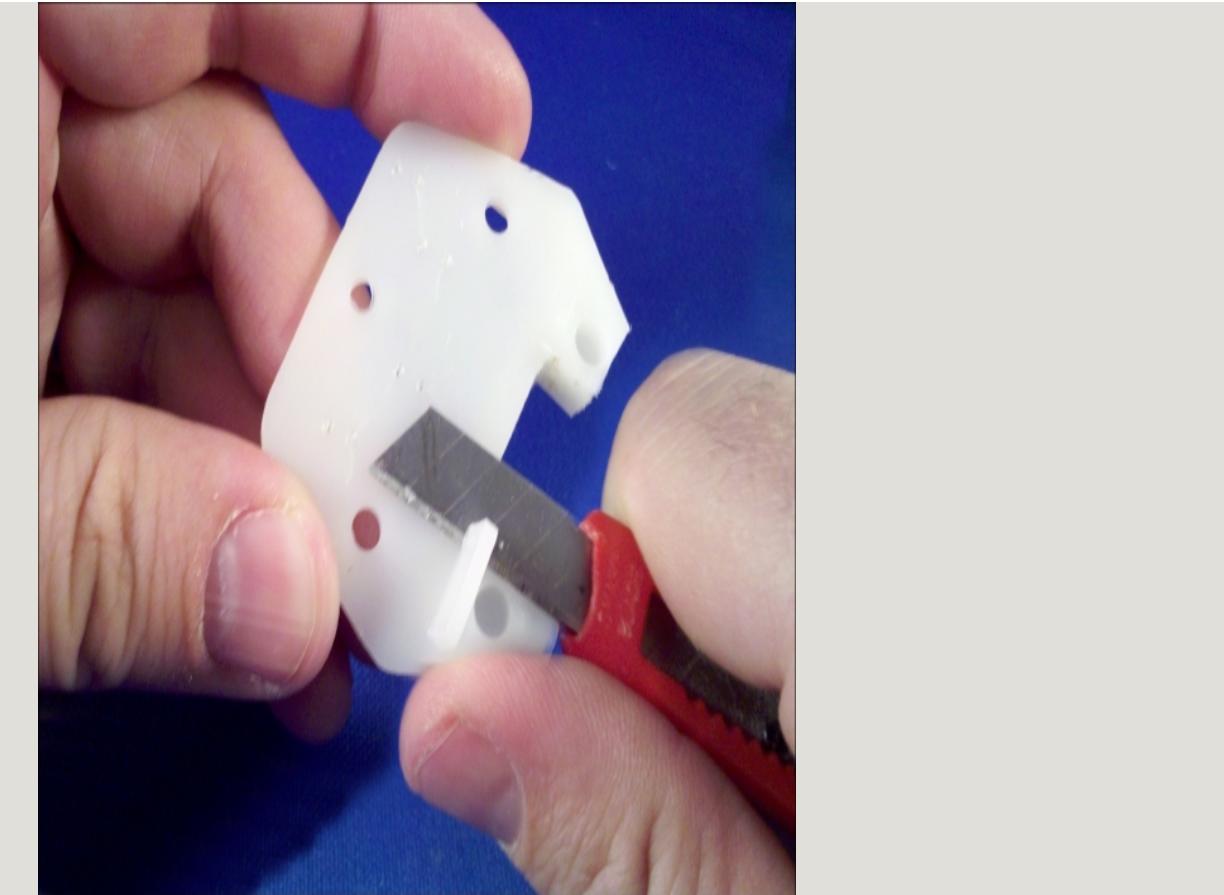
Then, to be sure that the surface is perfectly flat, use some fine-grit sandpaper on a flat surface to sand down the flat side:
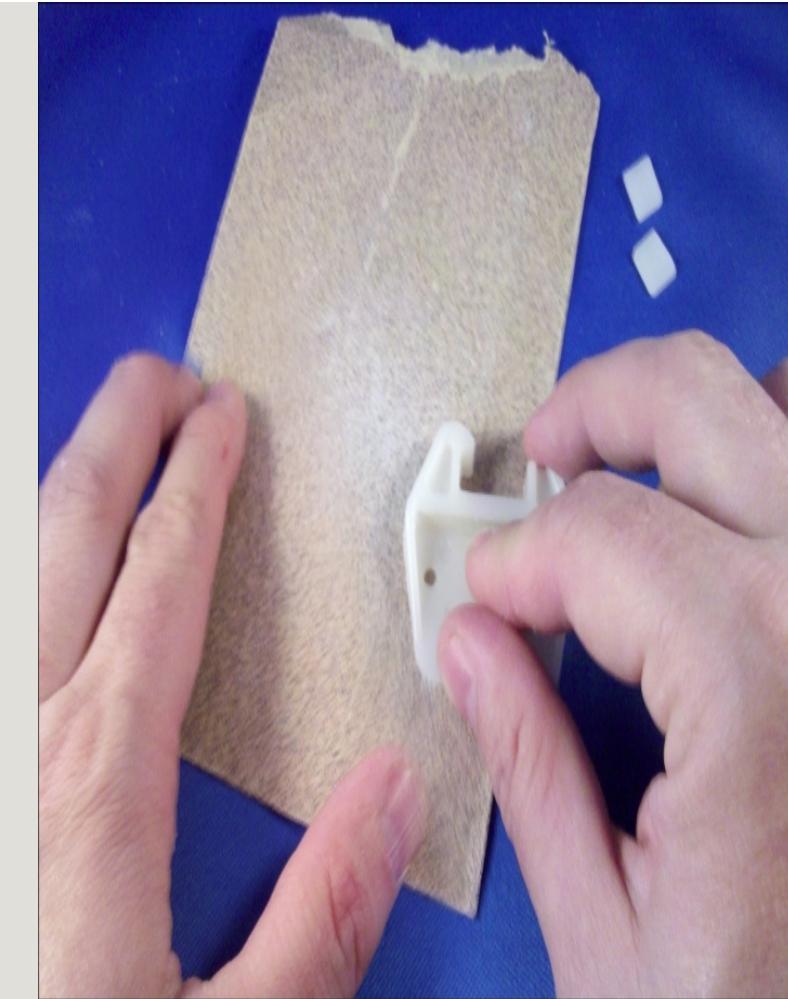
Next, take some grid paper, centimeter if available, and lay the object on the paper. Line it up with the grids as best as possible, and trace the object. Keep in mind that traces are generally a little bigger than the actual objects.

As an alternative, an ink pad can be used to make an even more accurate shape transfer. Just use the flat side of the object like a stamp. Again, try to line the part up with the grid as much as possible when making the impression in order to make the later modeling process easier. Stamping on a soft surface such as a towel can help get better coverage in the stamping process:

Finally, scan the stamped part in, or if a scanner isn't available, use a digital camera—whatever is possible to get the traced-image-on-graph-paper reference into the computer. Try to keep the grid lined up and even in the image, or some editing of the image will be necessary. It is not possible to edit the image in the modeling software beyond some basic transformations, so keep it as straight horizontally and vertically as you can and avoid skewing and warping of the image as much as possible:
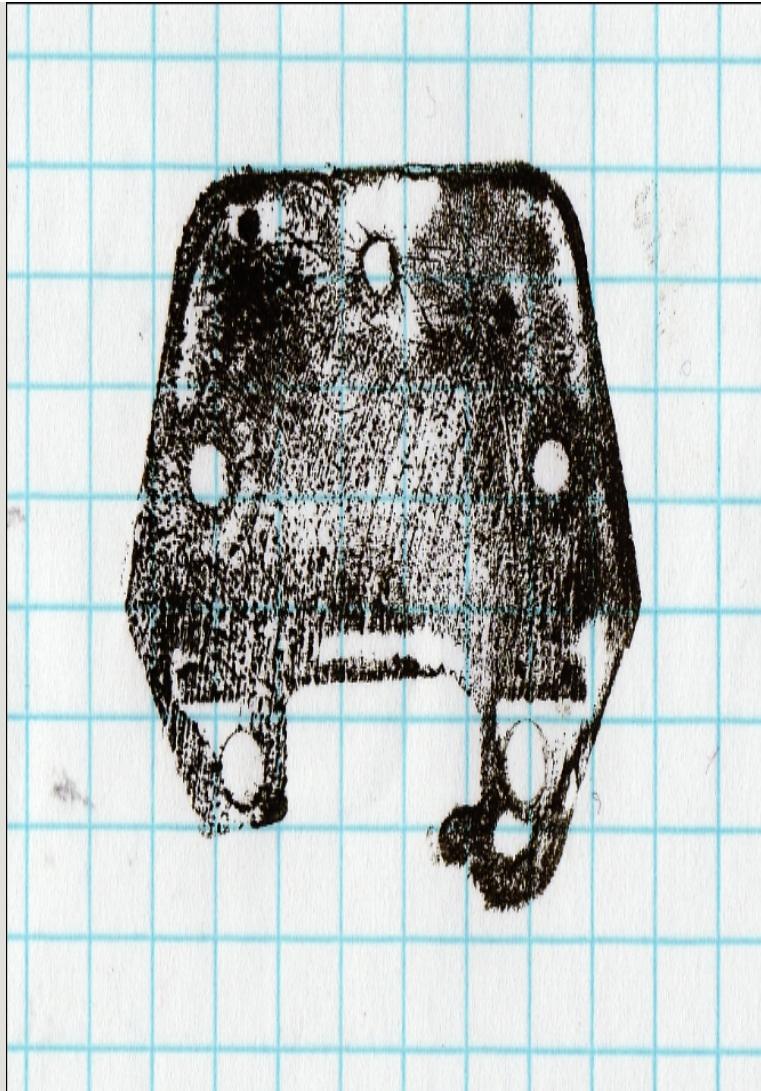
Importing the image into Blender

To follow along from this point, point your browser to http://thingiverse.com/thing:90754.

Now, find `Ch10 Scanned Image.jpg`, and right-click to save the image:

1. Start **Blender** and, as usual, clear the scene (*Ctrl* + *A* + *X*) and save it to start a project.
2. Give it an appropriate name in an appropriate directory, such as `Ch10 Measuring and Drawer Guide.blend`.
3. Change the view to **Top Ortho** (Numpad *7*, Numpad *5*).
4. In the **Properties** panel (*N*), locate the **Background Images** section, check the box next to it, expand it, and click on the **Add Image** button:
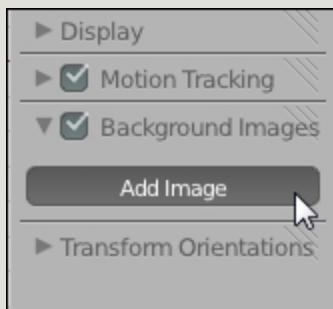
5. Click on the **Open** button and navigate to where the scanned image of the stamped object is stored. Then, open the image to place it in the scene:
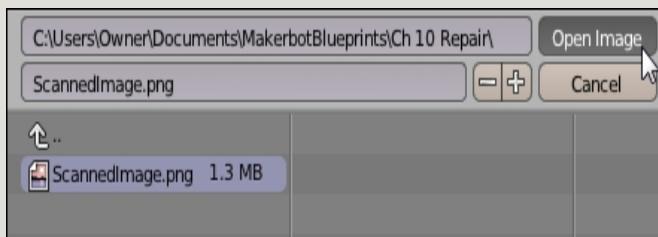
> **NOTE**
>
> The location and name of the scanned image may be different for you.

6. The first thing to note is that if the previous steps are followed carefully, despite the grid paper in the image

being a centimeter grid, at the default zoom, the grid lines seen are in millimeters. There are 10 millimeters per centimeter, so zoom out until the major grid lines start to appear, and then zoom out some more so that the millimeter grid lines disappear:
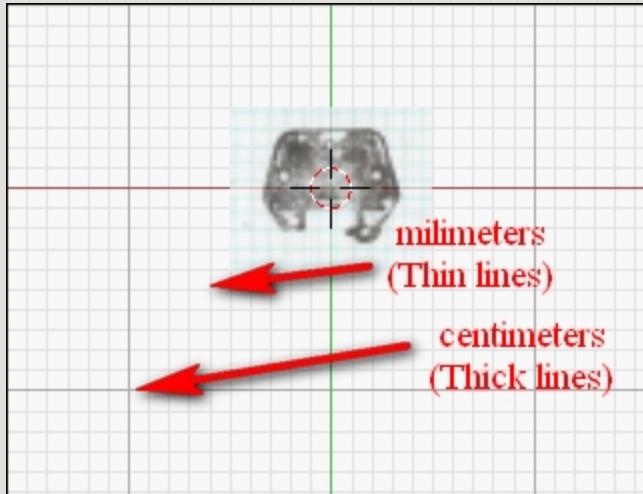
Adjust the settings in the **Property** panel for the background image until the grid on the paper lines up with the centimeter grid in **3D View**. A quick way to determine the necessary size is to count the number of grid lines in the scan, multiply it by 10, and divide by 2 (for the radius, which for some reason the size setting is). In the example, there are 12 grid lines shown, so setting the size to 60 gets the view very close to the right size.

7. Continue to adjust the **X**, **Y**, **Size**, and **Rotation** values as necessary until the grid lines in the image line up with the grid lines in the view properly:
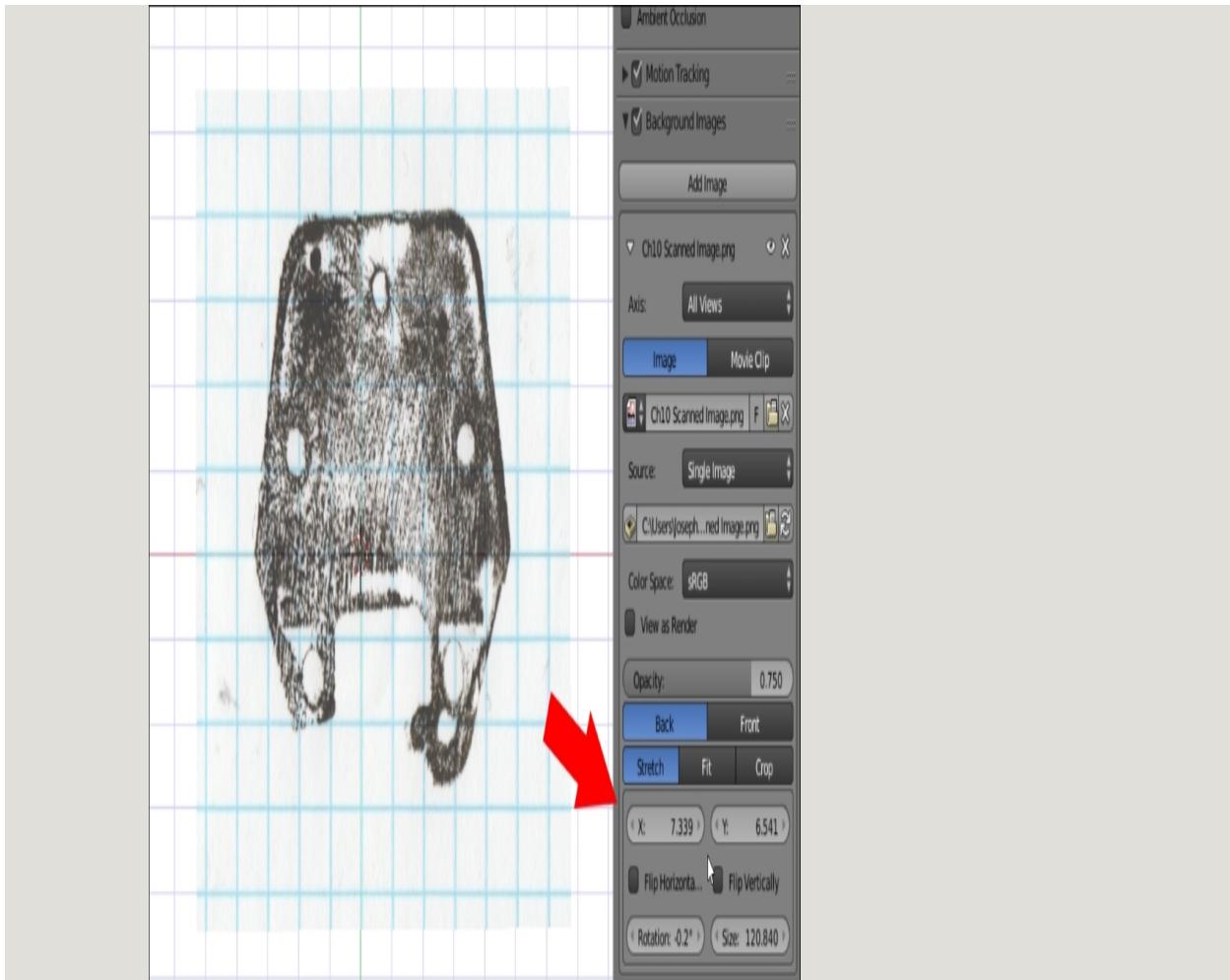
8. Since the object is to be symmetrical in this example, it would probably be a good idea to use the **Mirror** modifier, and to use it properly, it would be easier if the center line of the object were lined up with the world origin. The **X** value can now be adjusted to make it so. It no longer matters if the **X** grid lines line up with the image:
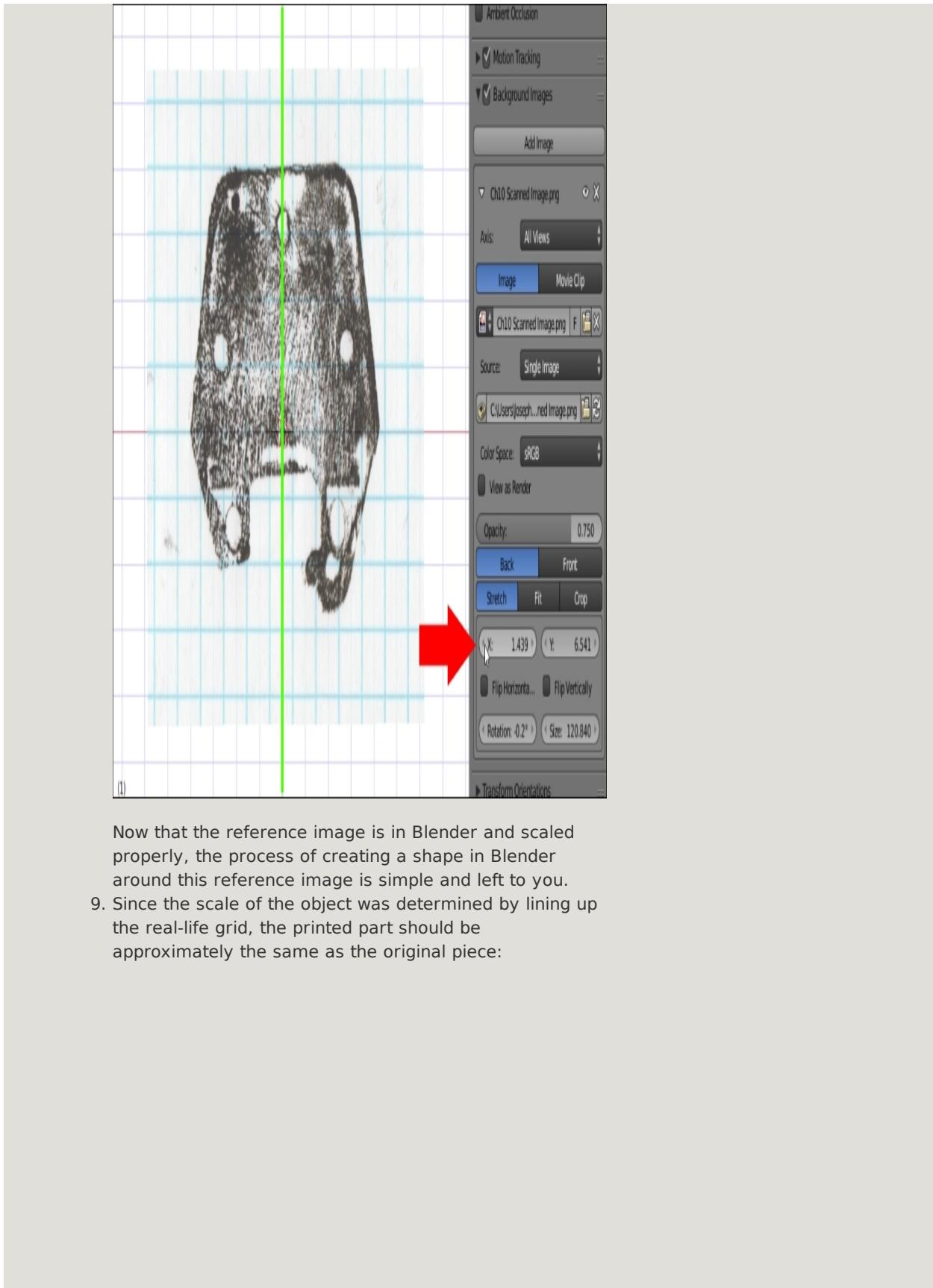
Now that the reference image is in Blender and scaled properly, the process of creating a shape in Blender around this reference image is simple and left to you.

9. Since the scale of the object was determined by lining up the real-life grid, the printed part should be approximately the same as the original piece:

The original is still a little messy after being used as a stamp, but the new one will take its place nicely.

Increasing reference pictures

In the background image settings, the pictures can be limited to specific views: top, bottom, front, left-hand side, right-hand side, and so on. Using these settings, multiple reference images can be combined to increase the accuracy of the model. If the shape is such that it's difficult to trace accurately from the side or front, it can be hard to trust the references. In these cases, the other images can be thought of as guidelines to be followed, rather than a reference to be trusted.

3D scanning

Finally, there is a method of capturing complex shapes that can quickly and accurately reproduce the shape on the computer, but at a considerably higher cost. There are many ways to 3D-scan objects.

Photogrammetry, or building a model from photographic references, captures the details of the shape of the object, but doesn't capture the scale very well. It captures small or big objects without distinction, but the resulting model can't be used for measuring purposes very well.

Structured light scanning can capture the scale pretty well if it's calibrated, but depending on a number of factors, the geometry may lack fine details. Often, scanners may cover that up by texturing the object, but textures don't print on FFF printers, which makes the cover-up obvious.

Some 3D scanners can only scan small objects. Some can only capture larger objects, but without any degree of small detail. Some require considerable effort on the part of the user to achieve any results. And if a 3D scanner does offer high detail and measuring accuracy, chances are it comes at a cost prohibitive to home users.

3D printing has driven up demand for accurate and cheap 3D scanners. There are many options available and new ones being developed all the time. As of the time of writing this, there is no clear family of winners that can be pointed to yet. Be cautious. Many promises are being made;

not as many promises are being delivered on. But keep an eye out because, one day, a reliable, accurate, and affordable 3D scanner is sure to be made.

Summary

Truth is, there's not much practical point to 3D printing if it can't be used to make real-life objects that conform to real life. Making measurements and applying those measurements to virtual objects is mandatory in 3D design. A ruler will do the trick in some cases, but more complex objects will require more robust tools. Calipers are a designer's best friend most of the time. Tracing the object onto graph paper can help transfer a complex shape into the computer, where it can be recreated. Finally, for the professional, a 3D scanner can quickly transfer a complex shape, but at considerable cost. The more options you have available, the better designer you will be.

In the next chapter, we'll use measurements to create a 3D model that will match a real-life object when we're done.

Chapter 5. An SD Card Holder Ring

Many 3D printers can print directly from an SD card instead of being hooked up to a computer all the time. This presents a problem of transporting SD cards without tying up your hands. Or pockets. A ring that you can put an SD card in is just the thing.

While a ring you can put an SD card in is a slightly silly solution, it's an excellent example of modeling based on physical objects. Both a finger and an SD card will need to be measured, and the design will have to be modeled too:
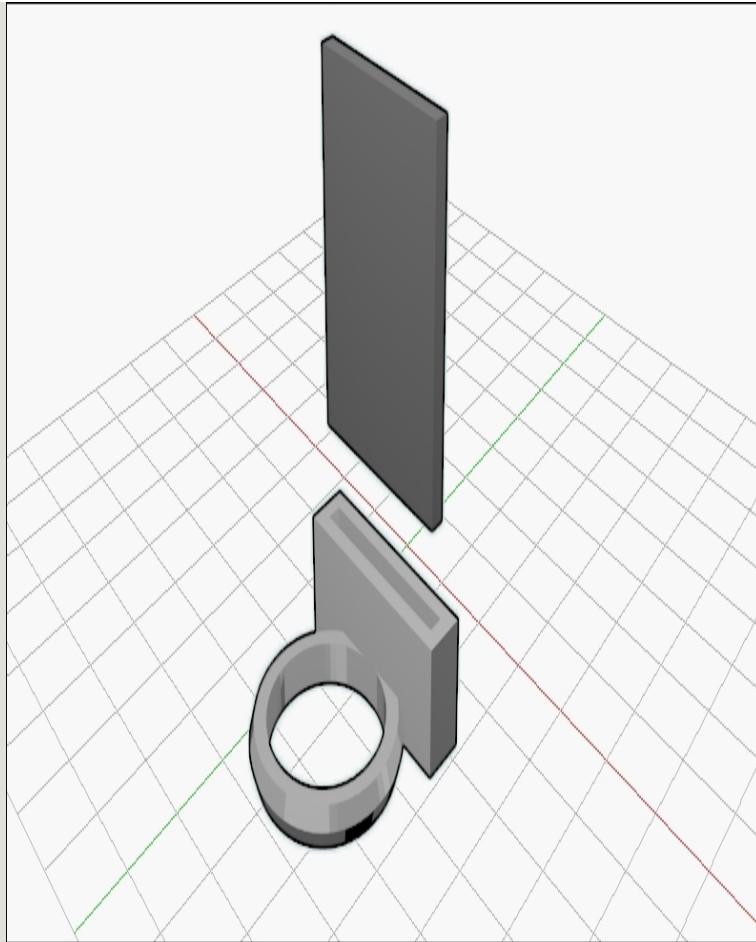
This chapter will walk you through the process of making a cool 3D-printed project, going through the following steps:

- Taking measurements
- Modeling the ring
- Testing the ring
- Adding an SD card holder
- Putting it all together
- Extra credit

The challenges of matching real-life measurements to 3D-printed objects is made difficult by plastic shrinkage and inaccuracies in 3D printers' operations. I'll comment on that when it becomes important.

Some would claim that Blender isn't capable of CAD-like precision, but this project will put those naysayers to rest.

Taking measurements

Before opening Blender, some measurements must be gathered. Careful measurements need to be taken of the ring finger and an SD card. A digital caliper is an excellent tool to take measurements with, as shown in the following figure:

My middle finger measures 19.3 mm at the widest point. If a digital caliper is not available, another way of measuring your finger is to wrap a piece of paper around the finger, mark where it overlaps, and use a ruler to find your finger's circumference, like this:
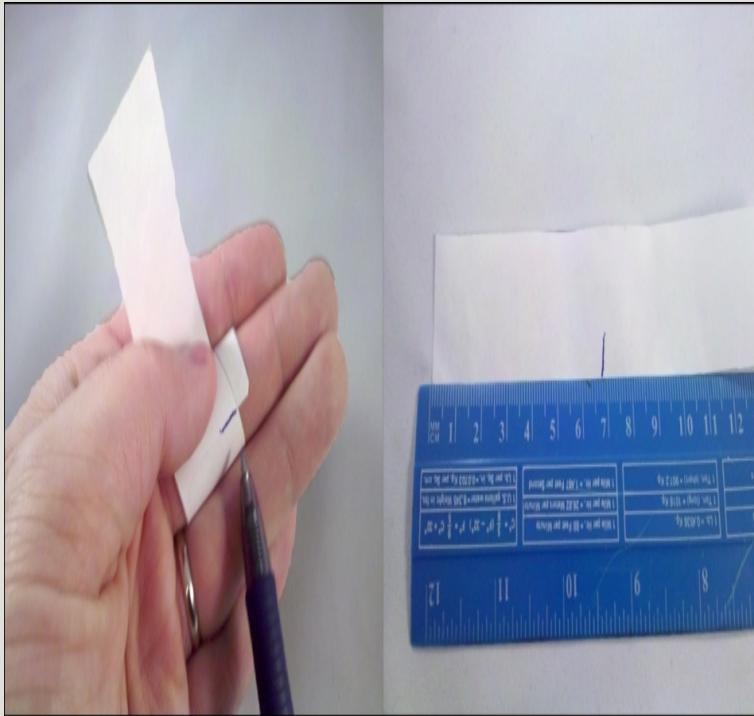
Then, consult the following table to find out the standard ring size and diameter:

Circumference (mm)	Circumference (in)	Diameter (mm)	Radius (mm)	US standard ring size
1.94	49.3	15.7	7.85	5
2.04	51.9	16.51	8.25	6
2.14	54.4	17.32	8.66	7
2.24	57	18.14	9.07	8
2.34	59.5	18.95	9.47	9
2.44	62.1	19.76	9.88	10
2.54	64.6	20.57	10.28	11
2.65	67.2	21.39	10.69	12
2.75	69.7	22.2	11.1	13

Circumference (mm)	Circumference (in)	Diameter (mm)	Radius (mm)	US standard ring size
2.85	72.3	23.01	11.50	14
2.95	74.8	23.83	11.91	15
3.05	77.4	24.64	12.32	16

An adult male is generally around a size 10 mm, an adult female is around a size 8 mm, and a child is around a size 5 mm.

The next thing to measure is a standard SD card. Fortunately, SD cards are all the same: 2.2 mm x 24 mm x 32 mm:

Modeling the ring

Now that all the measurements have been taken, it is time to go to Blender. Start a new scene and clear all the default elements (*A + A + X*). Then, save the scene (*Ctrl + S*) to create a new project in the directory of your choice. Name the project SD Card Ring.blend.

Modeling the finger

Let's start the modeling with the following steps:

1. Add (*Shift + A*) a Cylinder.
2. In the left-hand side bar, change the options for the cylinder. Change the **Vertices** value to 64 so the cylinder is smoother than default. Change the **Radius** value to half the measured finger diameter. Change the **Depth** value to 10:
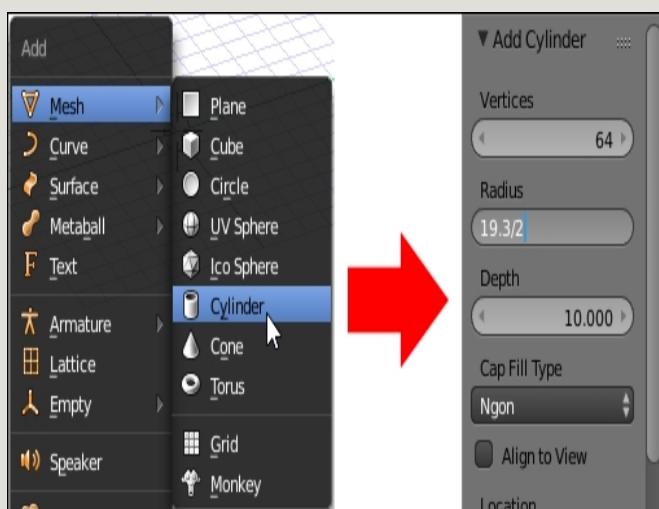
> **NOTE**
>
> Blender can take simple equations in these boxes, so instead of dividing the diameter by half to get the radius, simply type the diameter followed by /2, and it will divide it by 2 for you. So, in the example, typing 19.3/2 resulted in a radius of 9.650.

3. In the properties tab in the right-hand side bar, in the **Object** tab (orange cube), rename this cylinder to **Finger**, since it represents the finger in the build space:
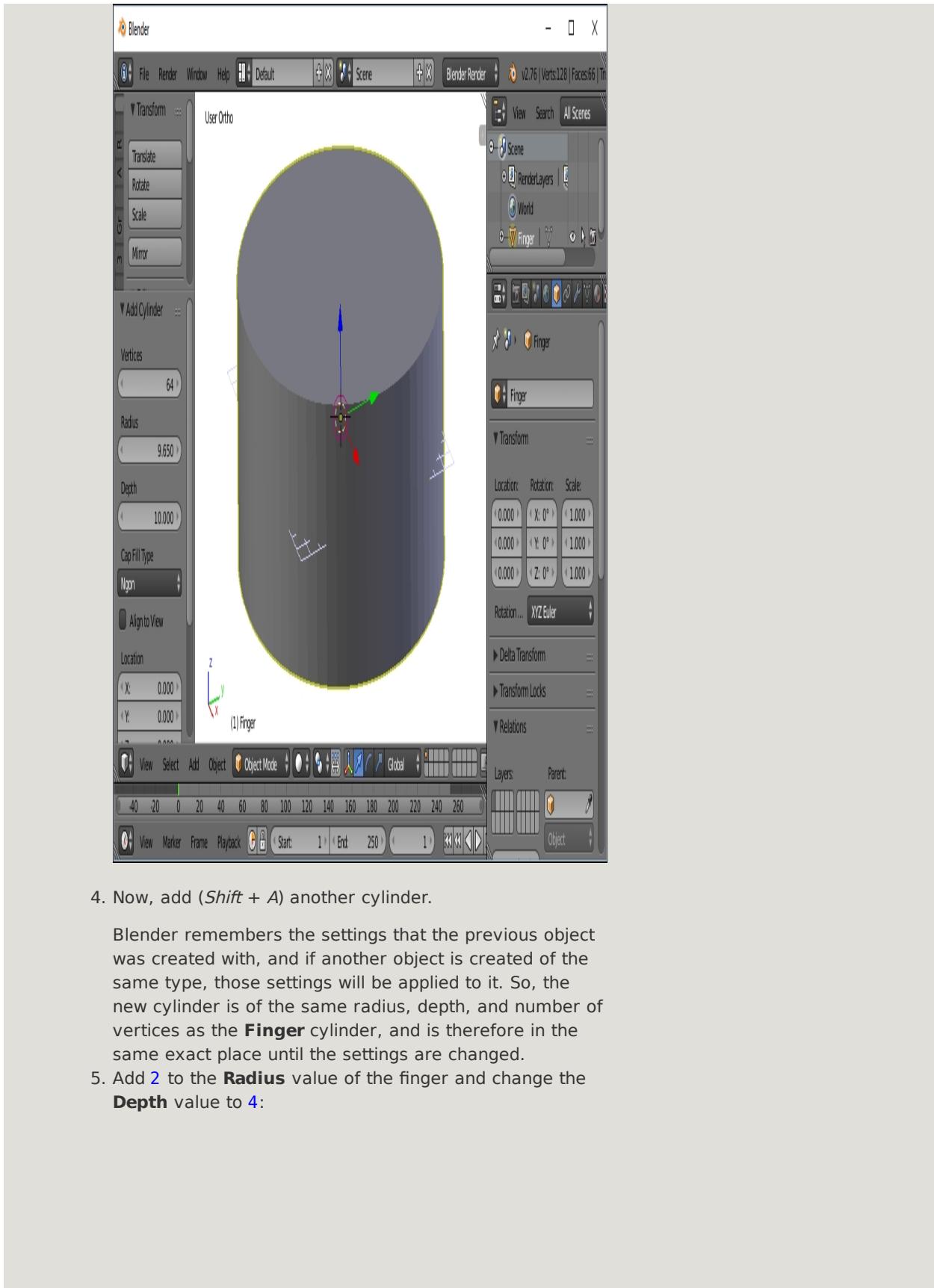
4. Now, add (*Shift* + *A*) another cylinder.

   Blender remembers the settings that the previous object was created with, and if another object is created of the same type, those settings will be applied to it. So, the new cylinder is of the same radius, depth, and number of vertices as the **Finger** cylinder, and is therefore in the same exact place until the settings are changed.

5. Add 2 to the **Radius** value of the finger and change the **Depth** value to 4:
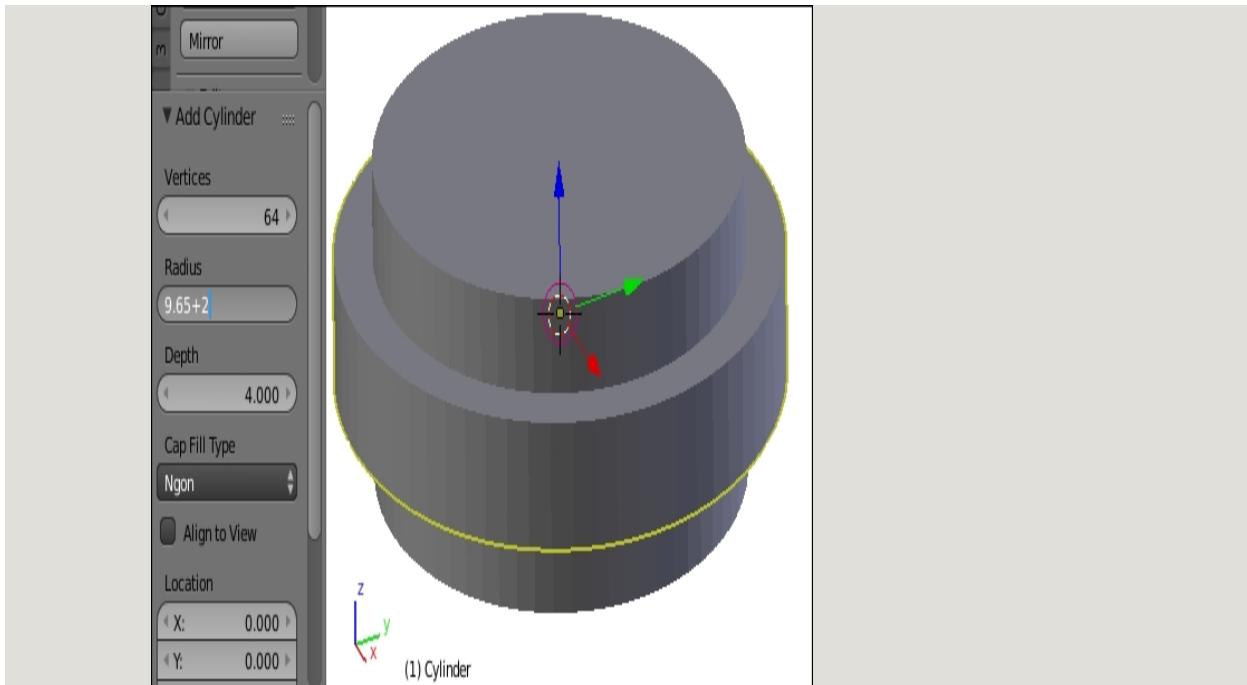
(1) Cylinder

NOTE

Again, adding 2 to the existing diameter is as simple as clicking on it and appending +2 to the end and then pressing *Enter*.

6. In the **Object** tab, rename this cylinder to Ring.

NOTE

The **XY** plane is the logical floor of the build space and passes through the origin. Blender places a (small) **XY** plane in the view space to help with visualization. It is, of course, completely imaginary, and objects can be built through this floor. In fact, by default, new objects are placed somewhat under the **XY** plane. However, respecting the **XY** plane can provide a solid base of reference for building objects if need be, especially when multiple objects are to be placed with precision.

7. Now, jump to the front (Numpad *1*) orthographic (Numpad *5*) view. You will notice that the ring is sitting half above and half below the origin:
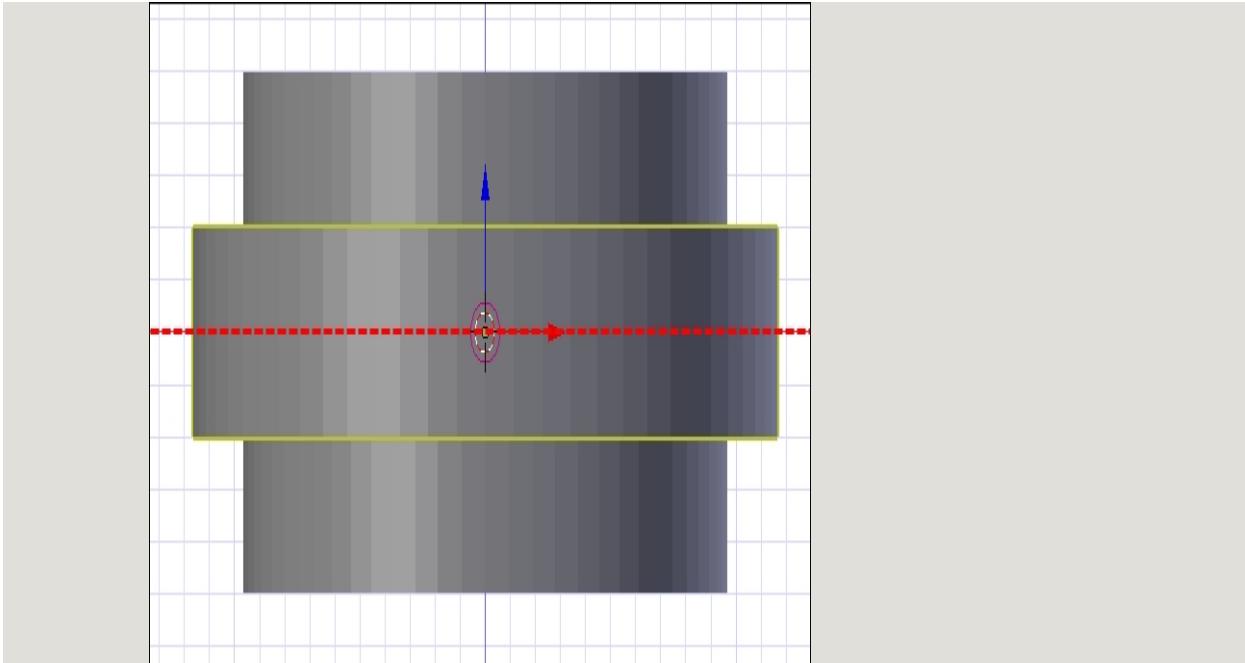
Putting the ring on the floor

By default, objects in Blender are created centered at the origin, meaning a part of them is under the **XY** plane. The goal of this section will be to place the important objects so they sit on the **XY** plane without going below it. This way, we always know where the bottom of the object is and can line everything up to this standard. So, with only the ring object selected, begin the grab (*G*) operation. Press *Z* to lock the ring's movement along the Z axis. Type 2 to move the ring up half its depth. Finally, press *Enter* to end the grab operation:
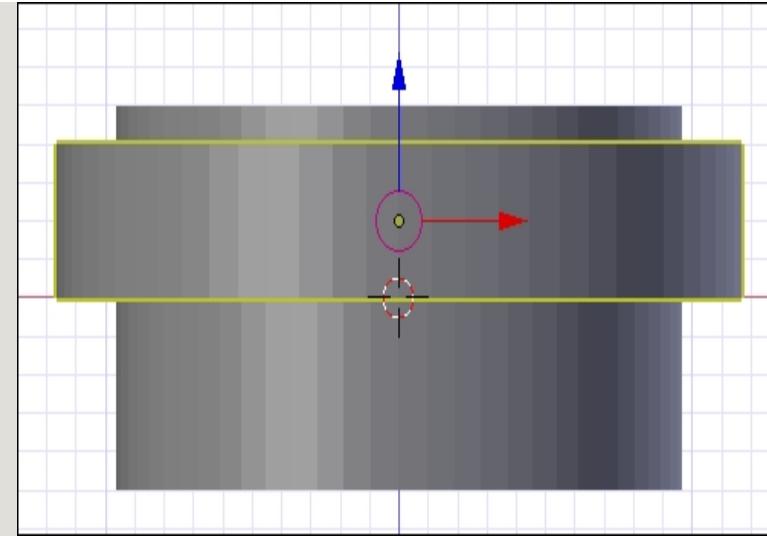
The finger cylinder is at an odd place in relation to the ring. But since the finger cylinder is just there to help us design the ring, and since it's both above and below the ring cylinder, it can just be left where it is.

Finishing the ring

Perform the following steps to complete the ring:

1. With the `Ring` object selected, add a **Boolean** modifier to it:
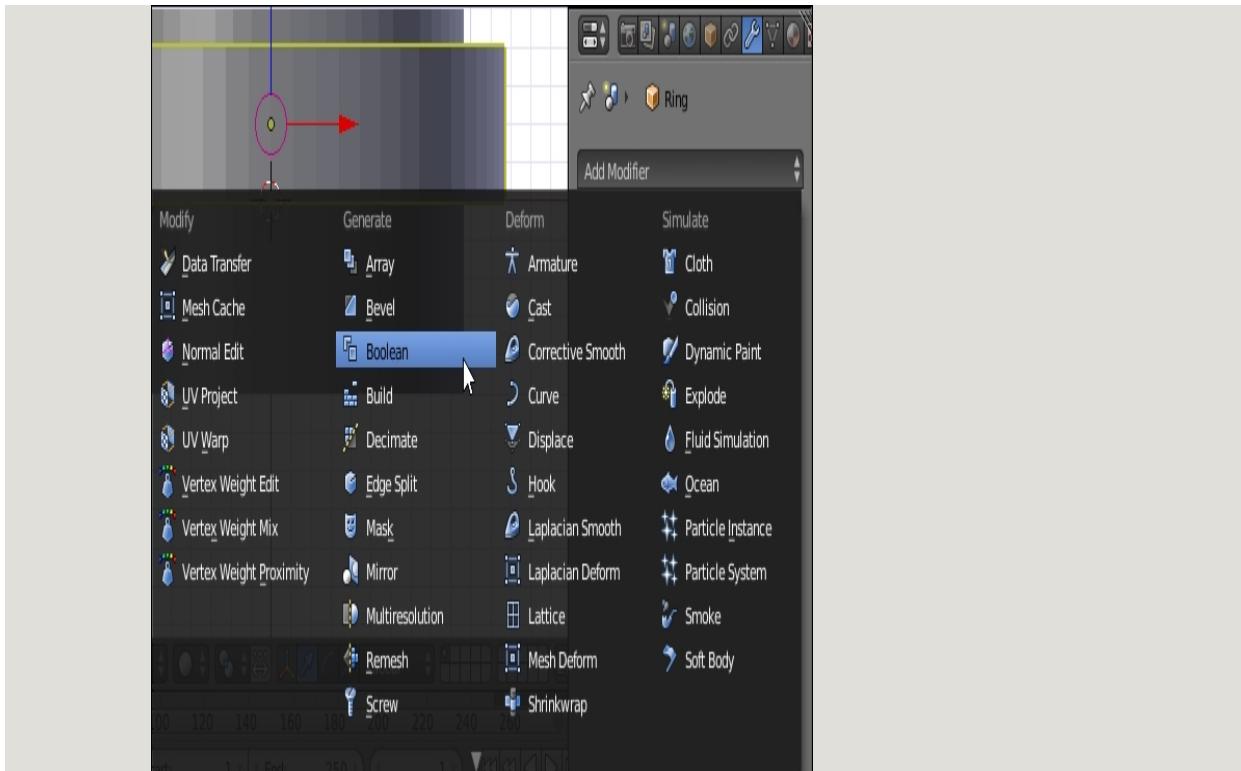
2. Change the the **Operation** field of the **Boolean** modifier to **Difference** and the **Object** field to **Finger**:
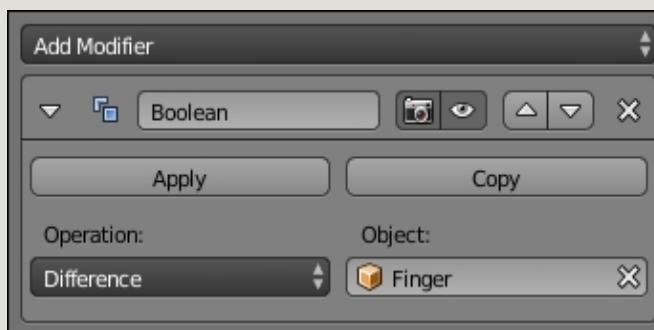
The **Finger** object is obscuring the effect of this operation, so it should be hidden. This may take the object out of the **3D View**, but it can still be selected in the **Outliner** panel and modifications can be made that way. Some editing remains impossible, however, until the object is unhidden.

3. Select the **Finger** object and, from the menu at the bottom of the **3D View** panel, navigate to **Object** | **Show/Hide** | **Hide Selected**, or press *H*. Another way is to click the icon that looks like an eye next to the **Finger** object in the **Outliner** panel:
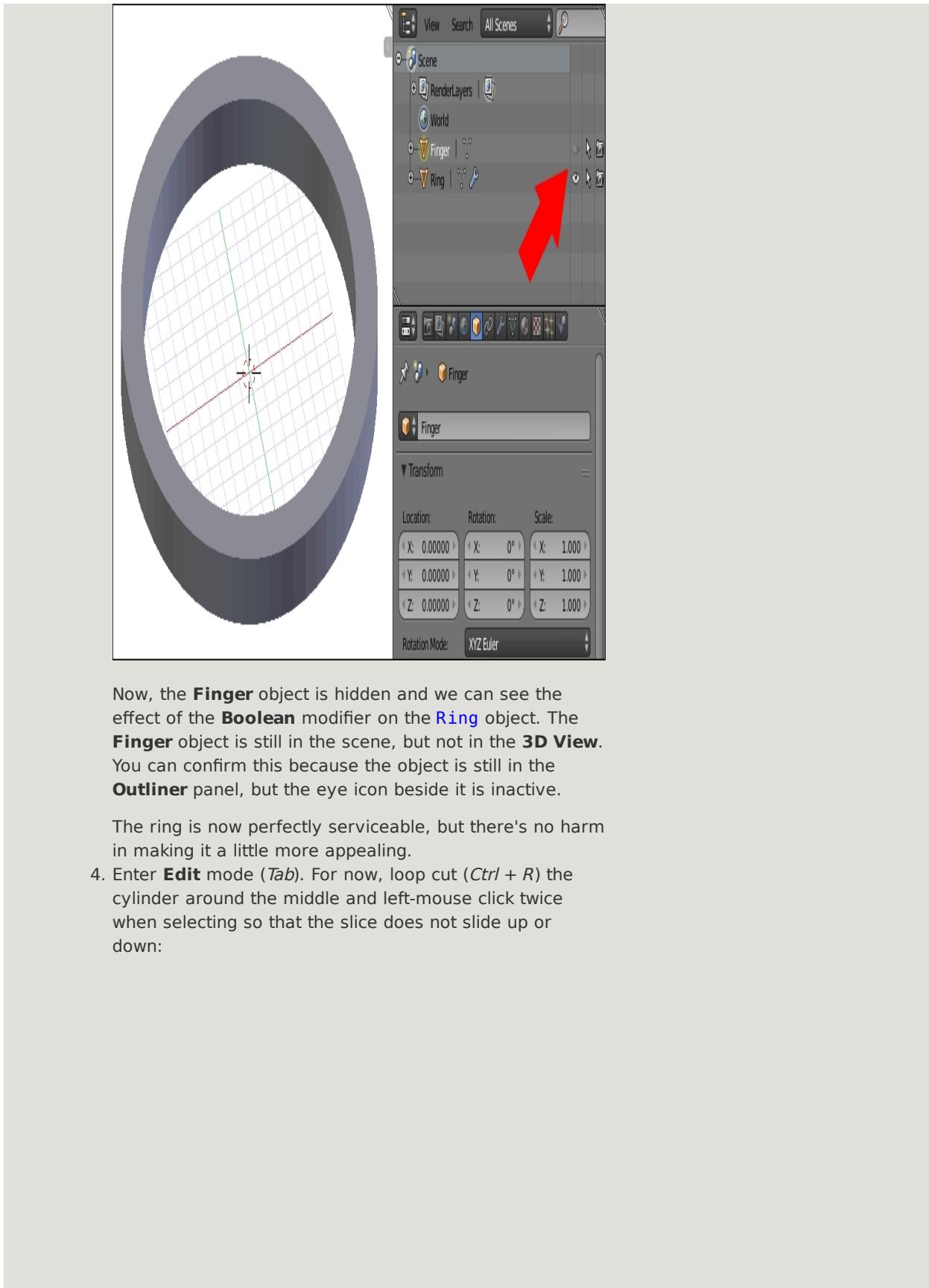
Now, the **Finger** object is hidden and we can see the effect of the **Boolean** modifier on the `Ring` object. The **Finger** object is still in the scene, but not in the **3D View**. You can confirm this because the object is still in the **Outliner** panel, but the eye icon beside it is inactive.

The ring is now perfectly serviceable, but there's no harm in making it a little more appealing.

4. Enter **Edit** mode (*Tab*). For now, loop cut (*Ctrl + R*) the cylinder around the middle and left-mouse click twice when selecting so that the slice does not slide up or down:
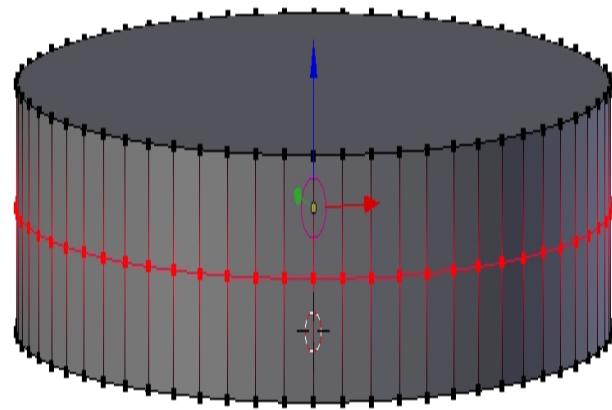
5. Scale (*S*) the newly sliced ring outwards slightly. The scaled ring of points should not extend more than 1 mm (one small grid square in the background) larger than the top and bottom disks of the ring shape.
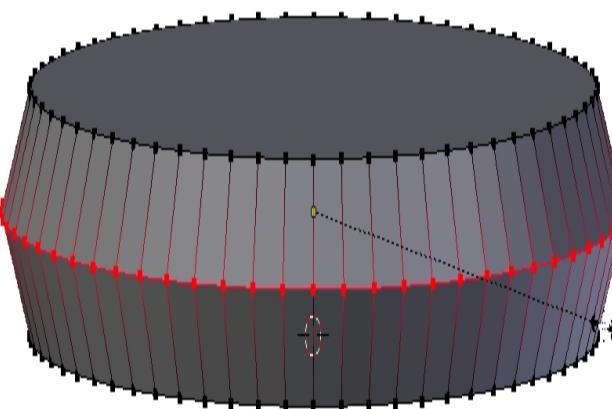
Scaling is an operation for which precise measured adjustments can be difficult to accomplish. This is one of the places where Blender falls a little short as a CAD-precise tool. Then again, this is a change made for aesthetics, so just eyeball the angle. Anything less than 45 degrees will be fine.

6. Now, exit **Edit** mode (*Tab*). Notice that the hole is back in the ring:
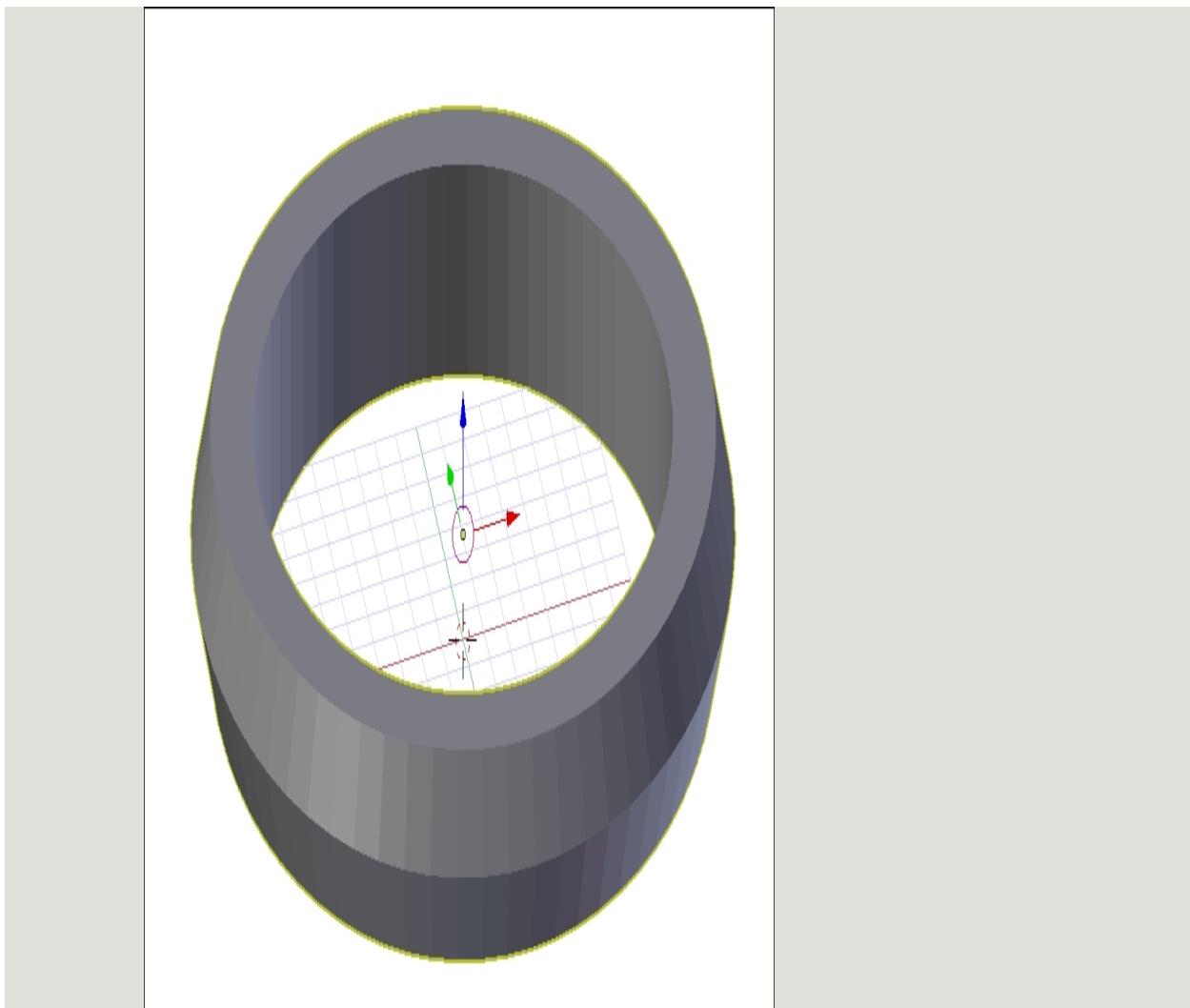
7. Remember to save (*Ctrl + S*) the project periodically.

Making a test print

If you are just learning to design without access to a 3D printer, then you can skip this step. However, since this print is so small, it is cheap, even through 3D-printing services. The ability to quickly make things for testing is one of 3D printing's greatest strengths, but it's not strictly necessary to learn how to design for them.

It's always good, especially when matching things to real-life objects, to test whether the fit is right. Some 3D printers print things smaller or larger than the file dictates. So, while the thing is small, and since 3D printers are so good at quickly testing things, test the ring you've made.

Export an **STereoLithography** (**STL**) of the ring through the **Export STL** option under **File | Export**. Name the exported file `Test Ring.stl`. Then, print the test ring and try it on:
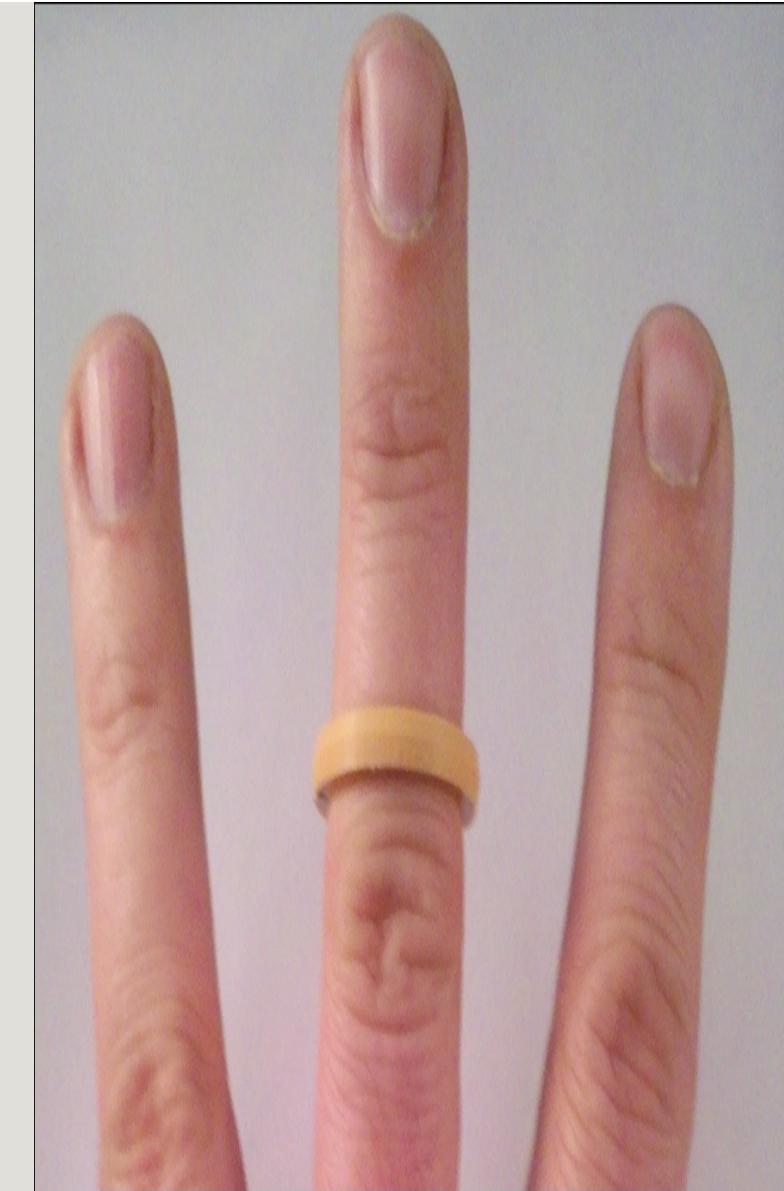
It might come as a surprise that the ring doesn't fit. If printing in ABS, the plastic may have shrunk when it cooled, as that is one of the properties of ABS. Or there may be a print setting such as filament diameter that isn't correctly set. Also, internal rings created by polygons are known to be a little small sometimes. Whatever may be the reason for the ring not fitting correctly, it's a good thing it was tested while the project was still simple.

Resizing the test ring

Back to Blender to make a new test ring. This time, instead of starting from scratch, reload the project and resize the existing ring. Take care to ensure that while resizing, the ring doesn't extend below the **XY** plane. So, to avoid this, we're going to resize the ring (and finger hole) from the **Properties** panel, like this:

1. Start by unhiding the finger. Do this by either clicking the eye icon for the **Finger** object in the **Outliner** panel or by pressing *Alt + H*:
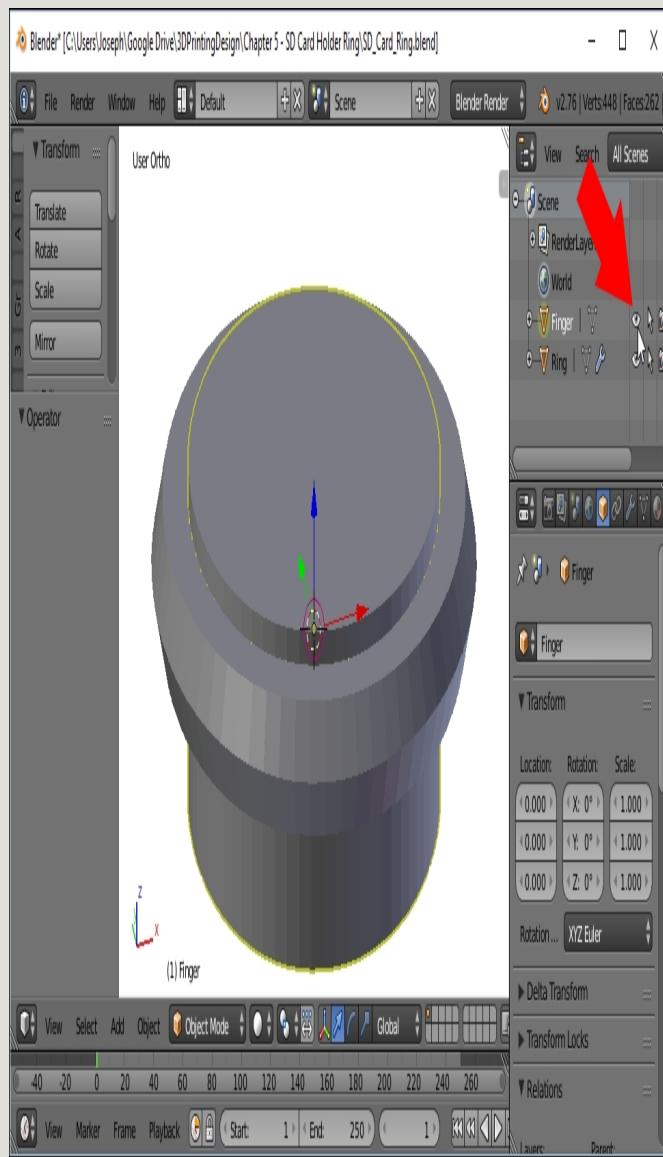
2. Select the **Finger** object and, in the **Properties** panel (*N*), locate the **Dimensions** area. Change the **X** and **Y** dimensions by clicking on the number to enter **Edit** mode and adding a +1 after the number shown. Leave the **Z** dimension exactly as it is.
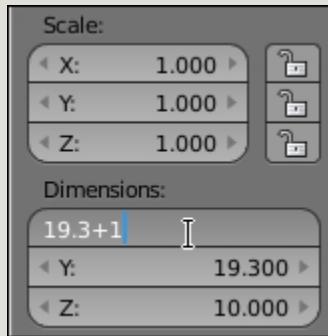
3. Now, select the **Ring** object. Similarly, change the **X** and **Y** dimensions to add 1 to them as well.

   This will make the finger hole 1 mm wider and the ring 1 mm wider as well. This way, the ring itself says the same thickness. If you don't feel as if 1 mm is enough added space, then add more to your ring. You can also make it smaller if it's too loose.

4. Save (*Ctrl + S*) and export the model again. Print it out and test again. Keep iterating and resizing the ring until a comfortable fit is achieved. When you've got it, hide the finger object and move on.

Adding an SD card holder

Now it's time to mount an SD card holder on the ring. The first step is to make a virtual SD card in Blender.

Organizing by layers

When working on a project with multiple parts in Blender, it's possible to organize things by using layers. While the idea of layers may not make as much sense in 3D as in a 2D graphics editor, the idea is similar. Shapes in different layers may share the same space, and they can be looked at and edited separately and independently. Layers are found in the menu at the bottom of the **3D View**, as shown here:
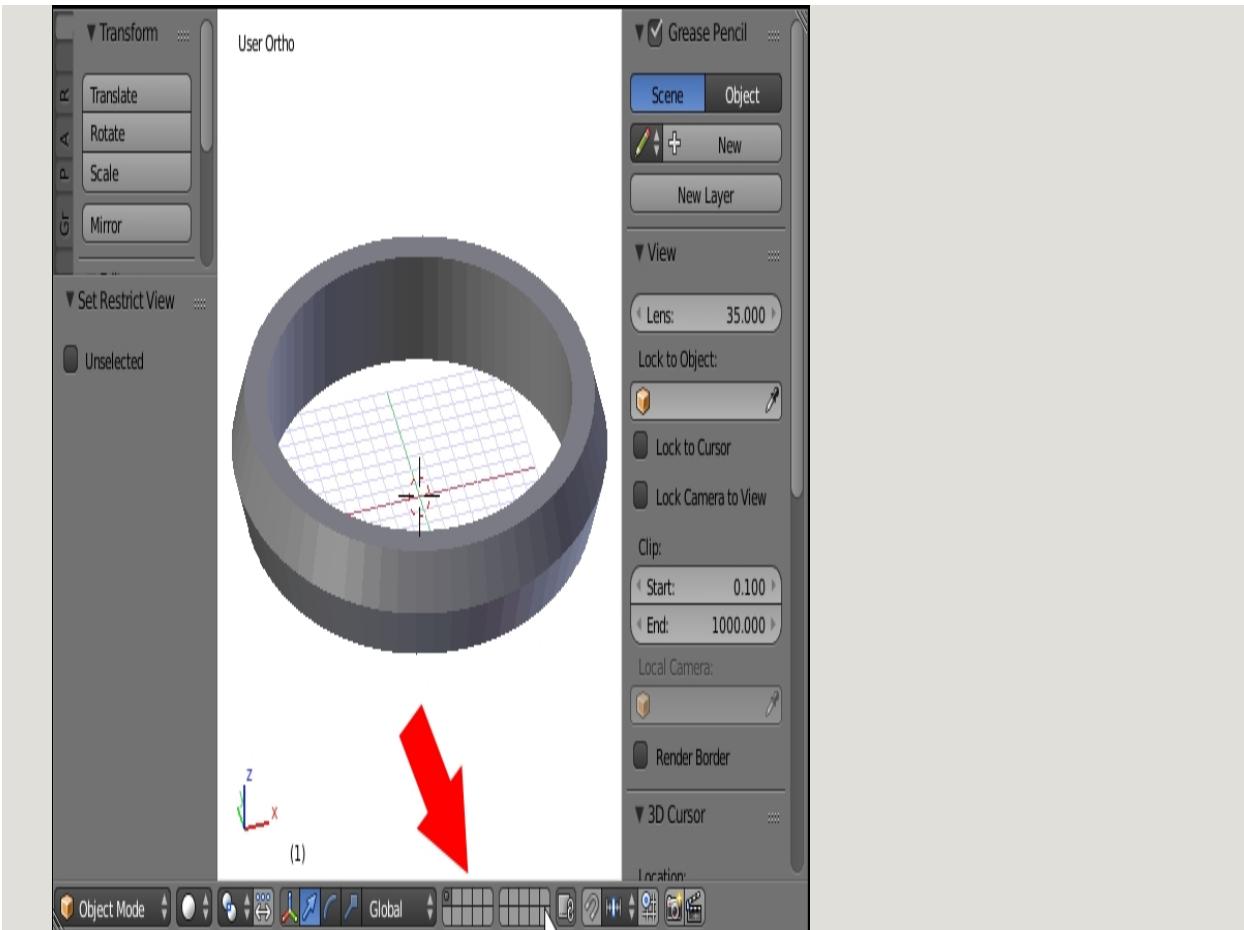
By default, the ring was created in the first layer, and there is a dot in the first layer to indicate that there is an object in it. To switch to an **Empty** layer, simply click on a square on the layer icons at the bottom of the **3D View** or type a number from the top row of the keyboard.

The number keys across the top of the keyboard (unless you're emulating the number pad) from *1* to *0* will switch to the corresponding layer in the top row. Holding *Alt* while pressing a number key will correspond to the rows in the bottom, for a total of 20 layers:

Now the scene looks empty, but checking the outliner panel, you can see that the ring and **Finger** objects clearly aren't gone. To see them again, simply click on the first layer with the object icon, or press the number key *1* at the top of the keyboard.

Now that you know the things you've created aren't gone, go back to the second layer.

Creating a virtual SD card

To create an SD card, perform the following steps:

1. Add (*Shift + A*) a cube to the scene.
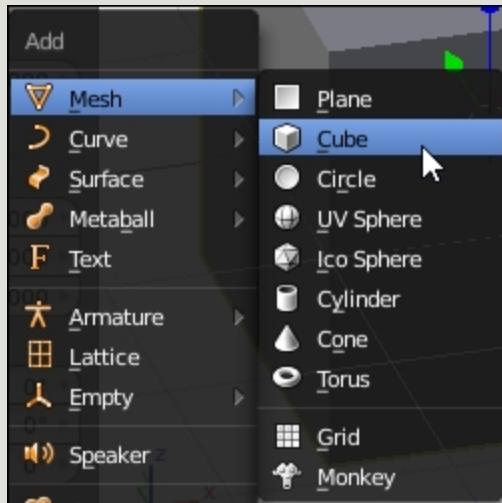
   We will be employing a trick to keep the cube above the **XY** plane as we change its size. Instead of moving the cube object up, we'll enter **Edit** mode and move all the points of the cube up. By doing this, the origin of the cube remains where it is and resizing the object will resize it around that origin. This will make more sense when seen practiced.

2. Enter **Edit** mode (*Tab*) and select all (*A*) points.
3. Then, grab (*G*) and move them 1 unit in the Z axis (*Z + 1*). Press *Enter* to end the grab operation:
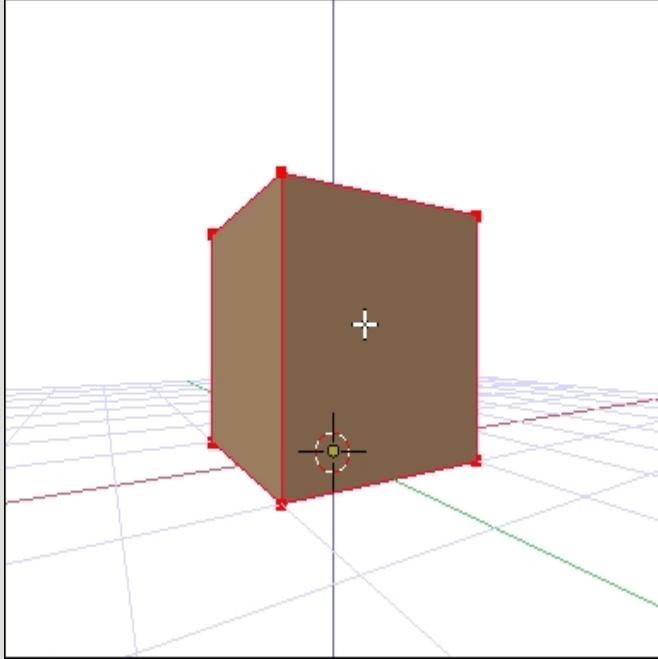
4. Now exit **Edit** mode (*Tab*).

   The cube now sits nicely on the **XY** plane, but more than that, its origin hasn't moved. So now, the bottom face of the cube is resting on the **XY** plane. Experimentally scale (*S*) the cube and move the mouse. Notice that with the cube's origin where it is, the scale operation is only scaling the cube above the **XY** plane. Undo (`Ctrl` + *Z*) any experimental scaling before continuing.

5. Now, resize the cube to the size of a SD card. To do this, in the **Properties** panel (*N*), locate the **Dimensions** area and change the dimensions to X: 2.2, Y: 24, and Z: 32:
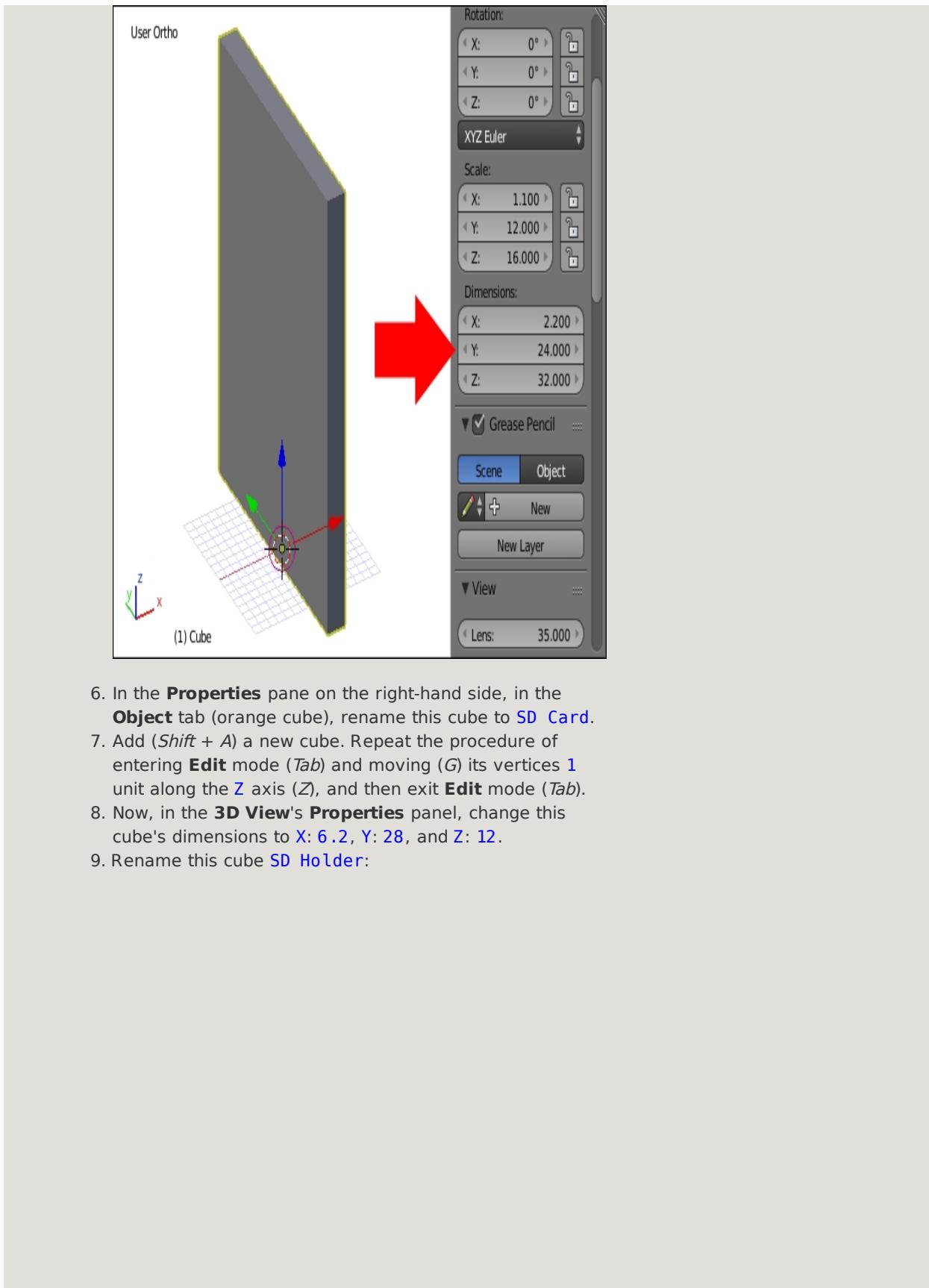
6. In the **Properties** pane on the right-hand side, in the **Object** tab (orange cube), rename this cube to `SD Card`.
7. Add (*Shift* + *A*) a new cube. Repeat the procedure of entering **Edit** mode (*Tab*) and moving (*G*) its vertices `1` unit along the `Z` axis (*Z*), and then exit **Edit** mode (*Tab*).
8. Now, in the **3D View**'s **Properties** panel, change this cube's dimensions to X: `6.2`, Y: `28`, and Z: `12`.
9. Rename this cube `SD Holder`:
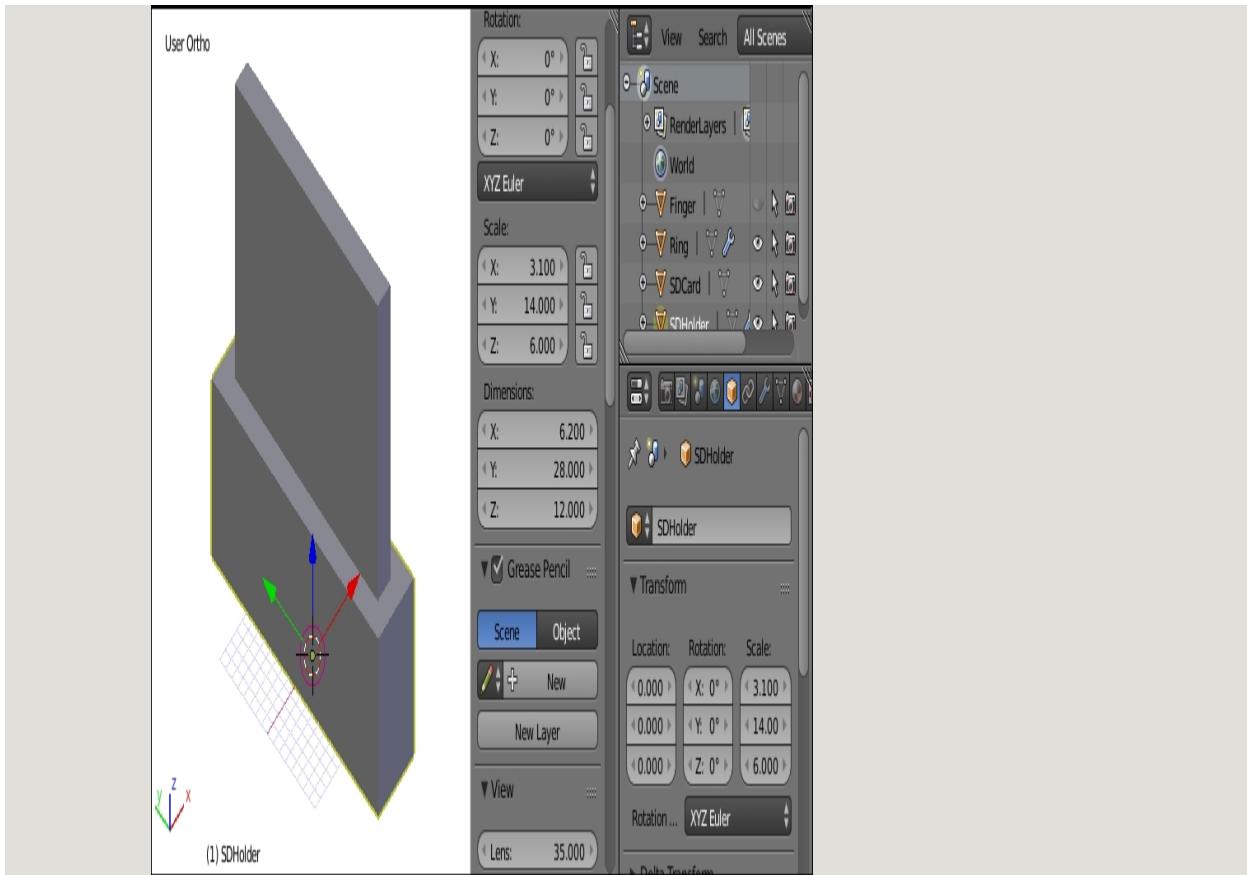
This makes the holder 2 mm thicker than the SD card all around, short enough to easily remove the SD card, while tall enough to hold it securely.

10. Add a **Boolean** modifier to the **SDHolder** object and **Difference SDCard** from it. Hide (*H*) the SD Card object and rotate the view to check the SD Holder object.
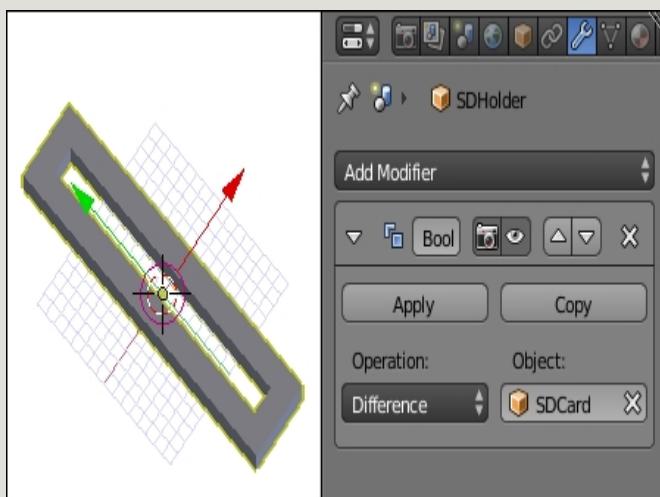
A hole all the way through was not the desired effect. The SD card might fall out! The Boolean operation has not

been applied yet, so changes can still be made.

11. Unhide (*Alt + H*) the SDCard object. Select it and move (*G*) it along the Z axis (*Z*) 2 units. Then, hide it again and check the SDHolder object.
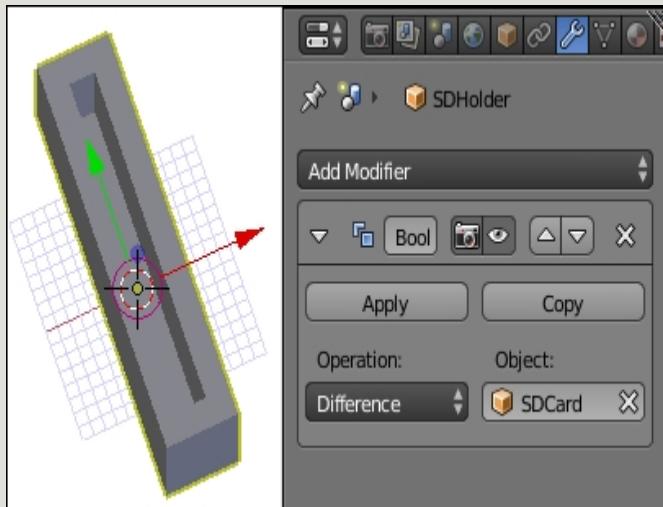
Much better!

Putting it all together

It's now time to bring the SD holder and ring together:

1. To start with, both the **SDHolder** and **Ring** layers need to be visible simultaneously. To make this happen, hold *Shift* while clicking on the other layer or typing the layer number with the number keys at the top of the keyboard:
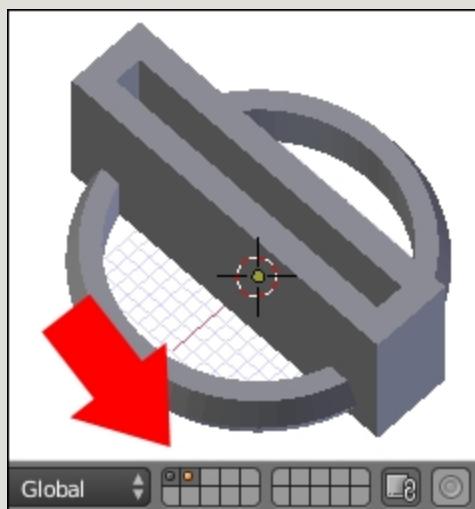
2. Select the **SDHolder** object and move (*G*) it along the X axis (*X*) until it is in front of the ring:

User Ortho

(1) SDHolder
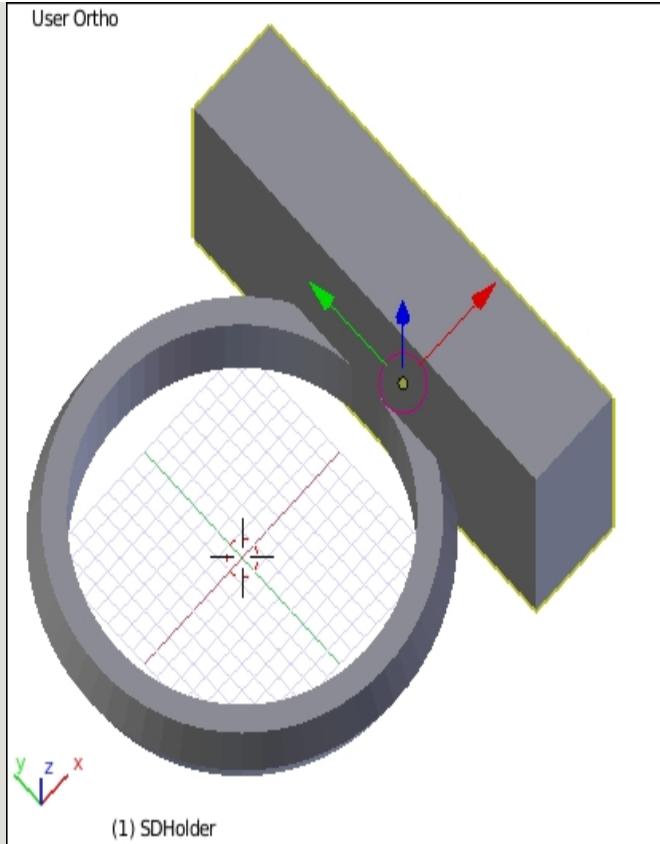
Depending on the view when the last operation was done, a problem may have been observed. What happened to the hole for the SD card? Well, because the **Boolean** modifier was not applied, the hole stayed right where it was, where the hidden **SDCard** object is. If you undo the last movement and redo the movement slowly, you can see the hole stay put as the **SDHolder** object moves.

User Ortho

(1) SDHolder
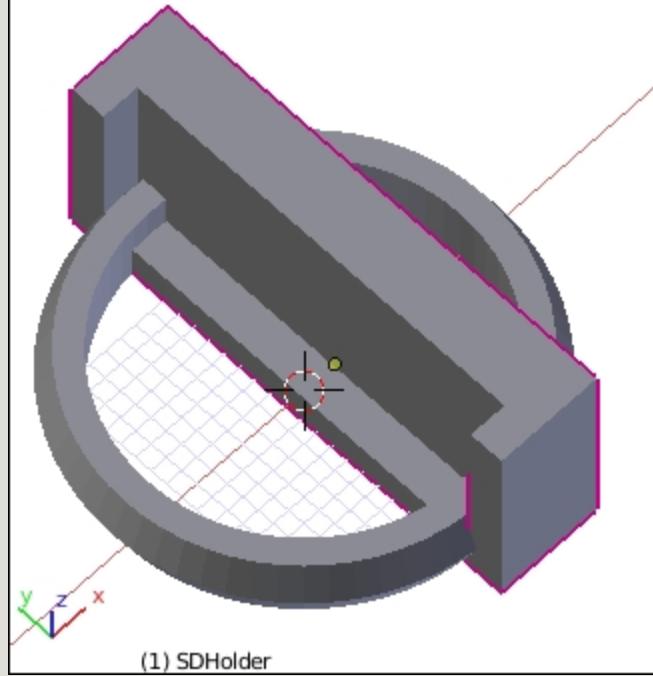
This is amusing but undesired behavior. There are two options to fix it. Either the **Boolean** operation can be applied before moving, or both objects can be moved together. And since there are plans for that hole that include cutting into the ring space, applying the **Boolean** modifier at this time is not the desirable step. So, the hole will have to be moved with the holder.

3. Undo (*Ctrl + Z*) the move operators until the hole is where it belongs in **SDHolder**.
4. Then unhide (*Alt + H*) the **SDCard** object. Select both **SDHolder** and **SDCard** by clicking on one and then holding *Shift* and clicking on the other.
5. Grab (*G*) and move them both in the X axis (*X*) until they are in front of the **Ring** object with ring intersecting **SDHolder**. Make sure there is a good connection here without interfering with the finger hole:

User Ortho
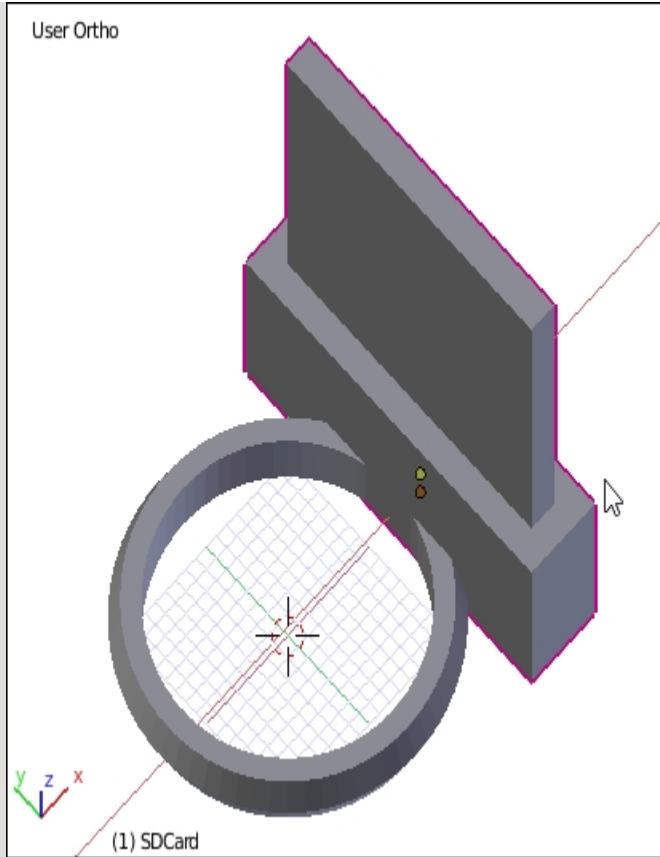
(1) SDCard

6. Hide the **SDCard** object again, now that everything is in place.
7. Select the **SDHolder** object and add a **Boolean** modifier to Union the **Ring** object to it.
8. Once the **Boolean** modifier has been added, hide (*H*) the **Ring** object to make inspecting the new object easier.

   There is a little problem with the **SDHolder** ring right now that is not immediately obvious. However, if seen from certain angles, for instance, looking into the hole from above, it's clear there's a blemish inside the hole. Inside the hole for the **SDCard**, the ring is protruding just a little. If this isn't clear, switch to **Wireframe** view (*Z*) and adjust the view.
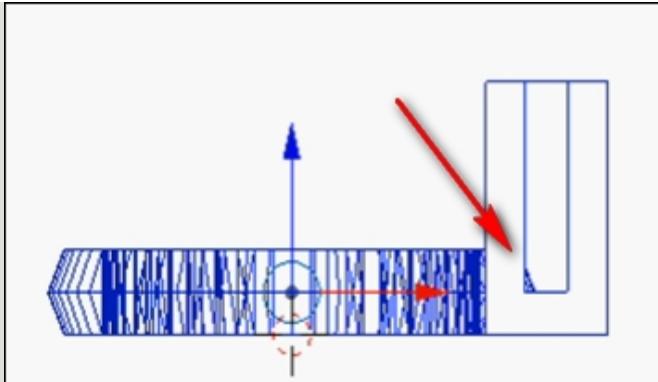
The solution is simple: instead of taking the SD card space out of the holder and then adding the ring to change around the order of the **Boolean** operations, the correct order is a Union of the **SDHolder** object with the **Ring** and then the **SDCard** object being differenced from the resultant shape.

9. In the **Outline** view, click on the **SDHolder** object.
10. With **SDHolder** selected, click on the move down arrow button on the first **Boolean** modifier to move it down the stack:
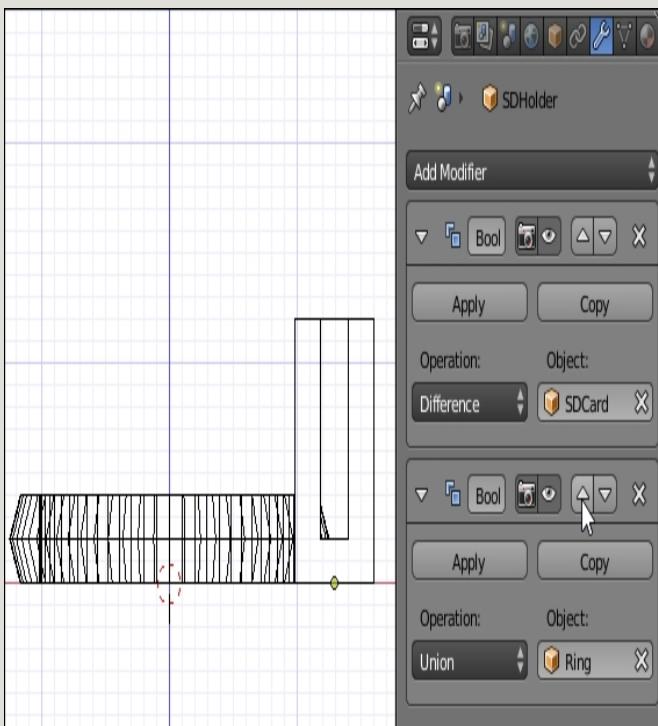
11. The **SDHolder** object is now complete and correct, with the ring attached to the holder and a complete hole for the SD card to sit in:
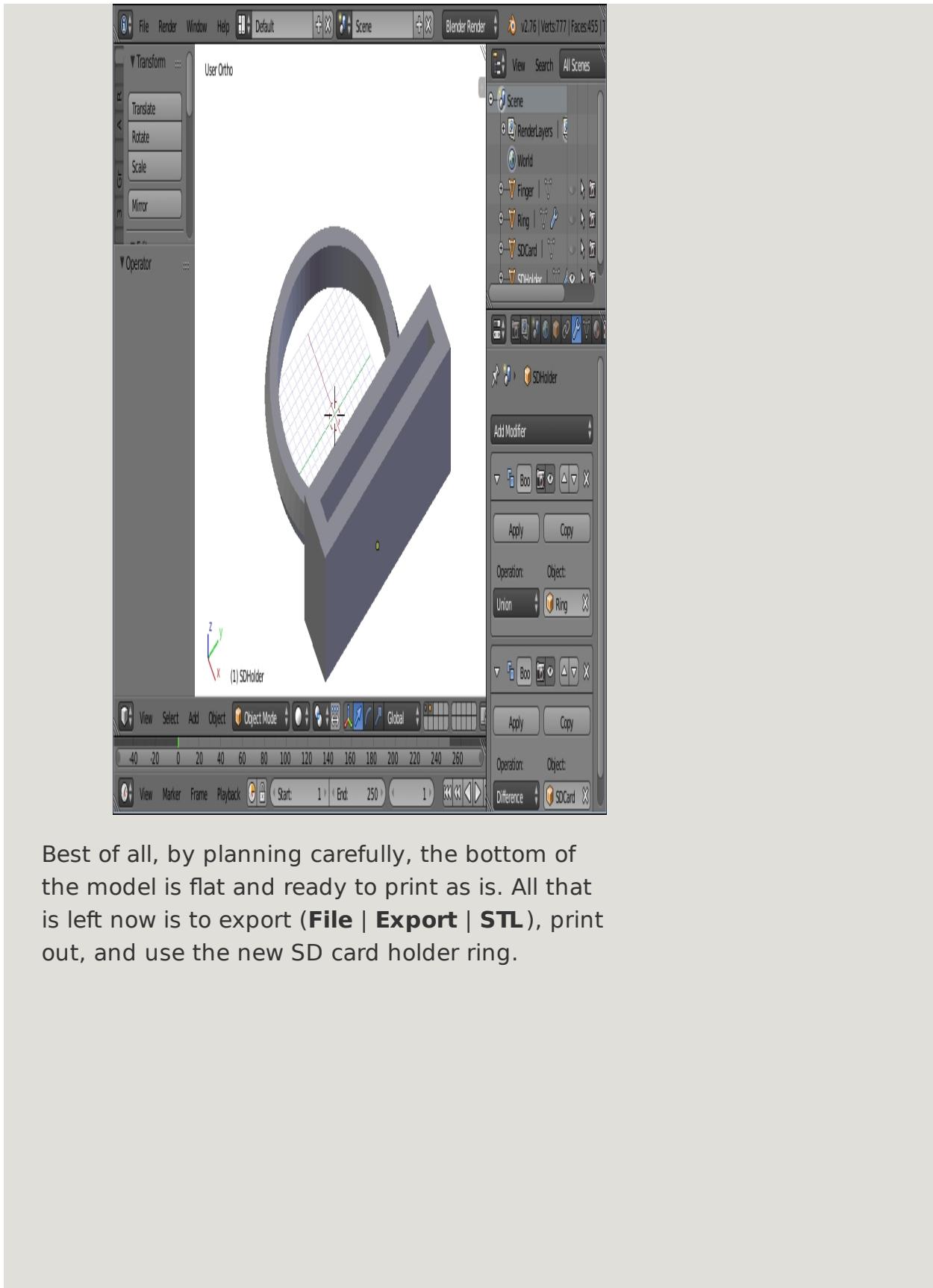
Best of all, by planning carefully, the bottom of the model is flat and ready to print as is. All that is left now is to export (**File** | **Export** | **STL**), print out, and use the new SD card holder ring.

Extra credit

In this project, modifiers were used on simple objects right up to the final object. This allowed the creation of a more complex object while retaining the editability of the simpler objects. However, leaving modifiers unapplied becomes undesirable when objects become too complex. Having to rebuild an object every time an edit is made can crash a computer.

For fun, try to see how quickly Blender will slow down, and maybe try to crash it, by making several cubes and adding a **Subsurf** modifier to increase their polygon count. Then **Boolean** the smoothed cubes together, but don't apply the **Boolean** modifiers. Add more and more **Subsurfed** cubes and add them to the **Boolean** object. Then start moving around and editing the cubes to see when performance becomes choppy. It may be surprising how little it takes, depending on your system:
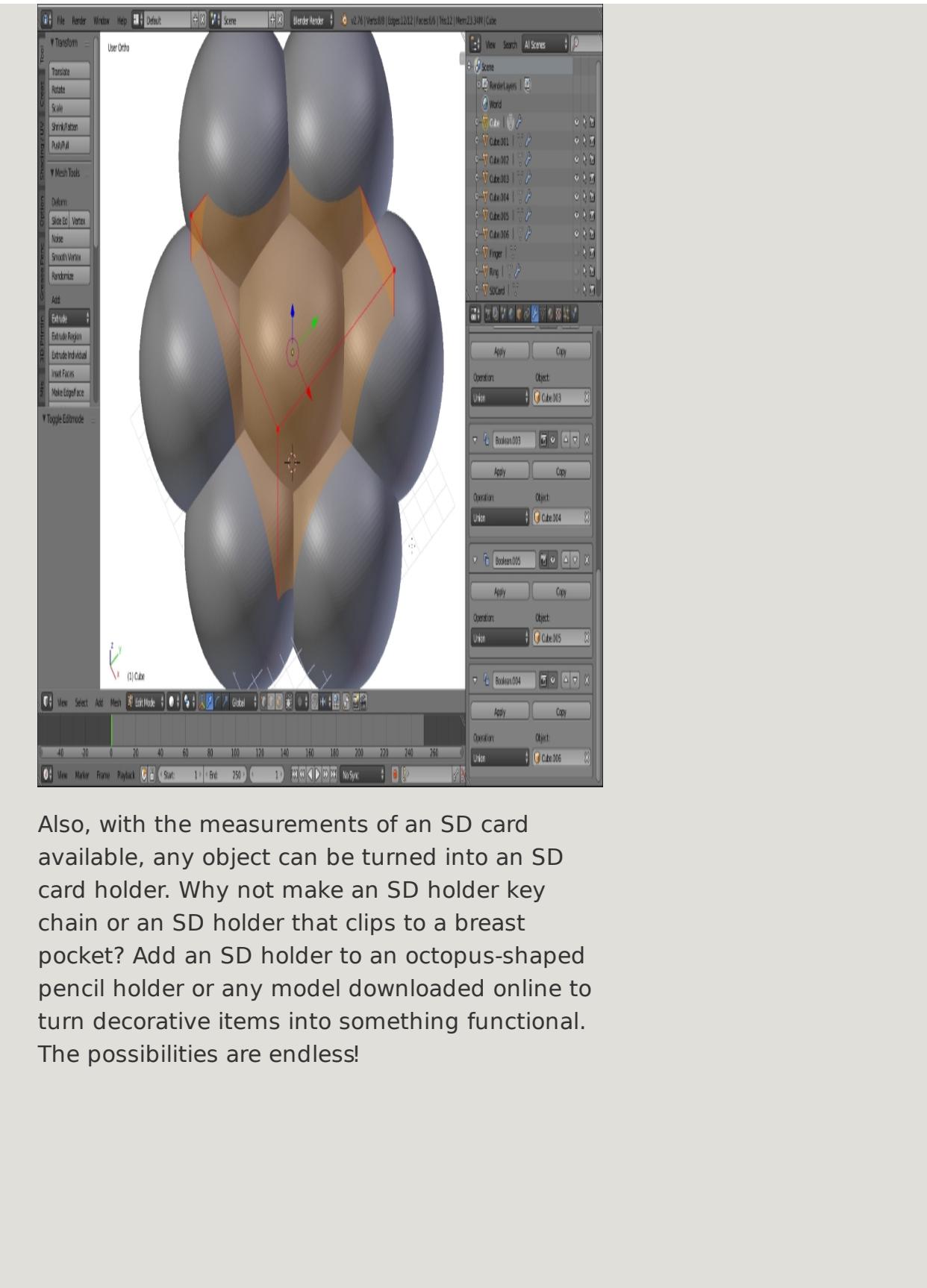
Also, with the measurements of an SD card available, any object can be turned into an SD card holder. Why not make an SD holder key chain or an SD holder that clips to a breast pocket? Add an SD holder to an octopus-shaped pencil holder or any model downloaded online to turn decorative items into something functional. The possibilities are endless!

Summary

Modeling things to fit real-life objects is one of the best and most common applications of 3D printing, and Blender is capable of this. Blender may never be able to compete with other CAD programs for precision; nevertheless, with some careful planning and clever manipulation, Blender is capable of extremely precise modeling.

Also, leaving modifiers unapplied is a technique that allows a high degree of flexibility, which is particularly desirable for something that stands a good chance of being customized. If someone else wanted their own SD card holder ring, it would be a simple matter to resize the ring and adjust to position the SD holder.

Now that you have the knowledge and skills to measure and design in Blender, the world can be at your fingertips.

Chapter 6. Sculpting the Face of the Sun

Sometimes, 3D printing needs to be precise and practical. And sometimes, it just needs to be fun, like the wood-cut puzzles your grandpa may have made in his garage. We'll be making similar puzzles but with a whole new dimension in order to take advantage of the power of 3D printing.

In addition to the basic 3D editing tools you've already learned about, Blender also includes some robust sculpting tools that enable editing objects with actions similar to drawing or building up clay. Because of its nature, sculpting facilitates a very organic modeling process.

There are a lot of sculpting tools available, and each one has a special use. The project in this book will use the most popular and useful of them to sculpt a simple model. This model will then be combined with basic editing outside of sculpt mode to create a 3D puzzle.

Creating the base object

Since the editing tools can only work on existing objects, the first thing to do is create a new object:

1. Start Blender and select all objects (*A*) and delete them (*X*).
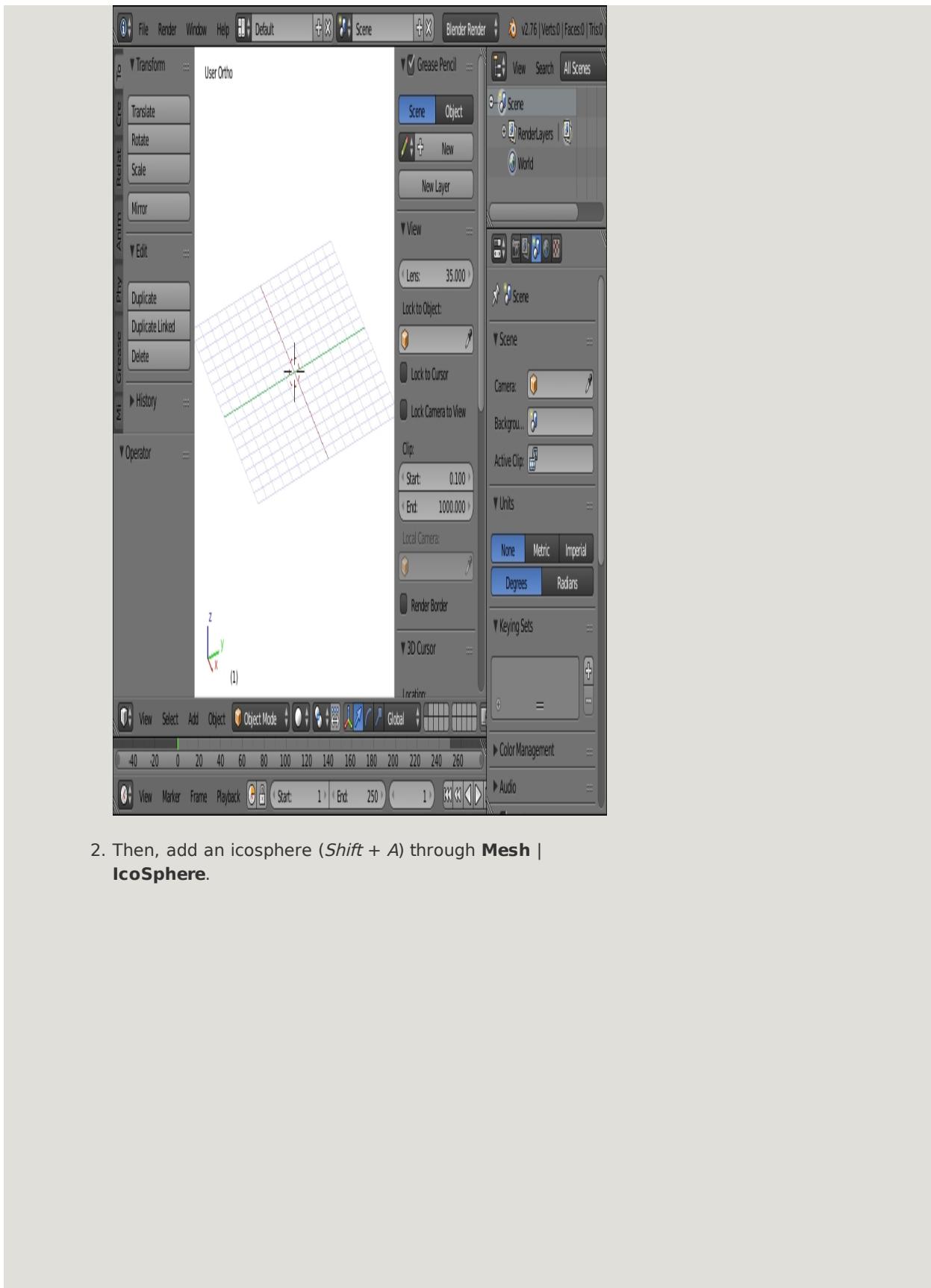
2. Then, add an icosphere (*Shift + A*) through **Mesh |
   IcoSphere**.
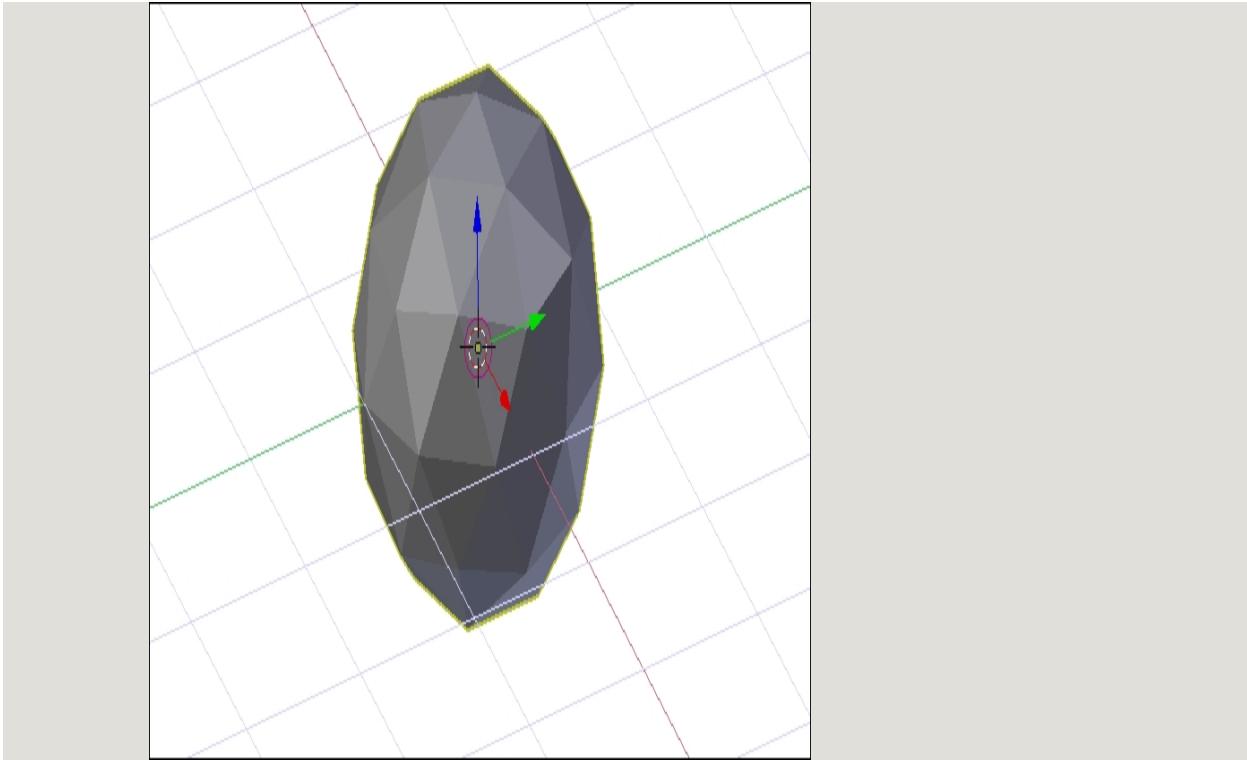
Icospheres differ from normal spheres in that they don't bunch up at the poles. This will work out better for sculpting. But if viewed from the top (Numpad *7*), the icosphere is not symmetrical from left to right. To fix this, follow these steps:

1. Rotate (*R*) the icosphere 90 degrees along the Y axis (*Y*).
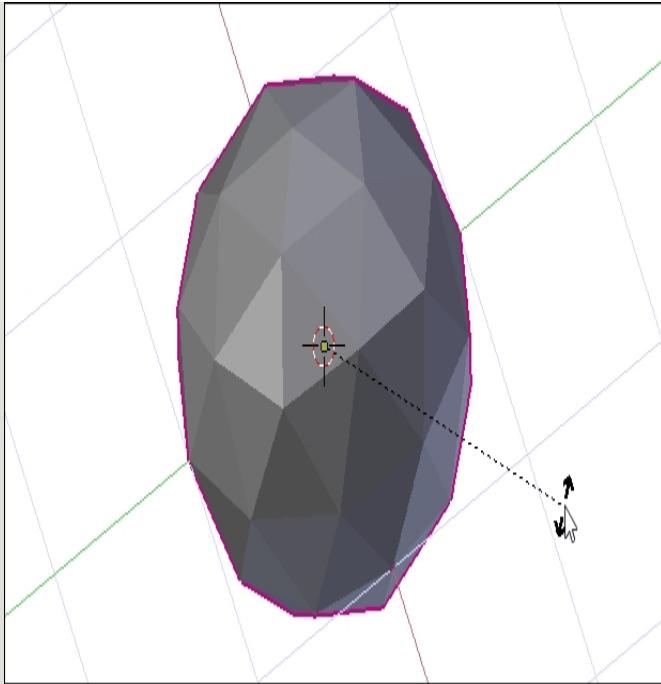
2. To make the icosphere into a disk, scale it (*S*), constraining to the X and Y axes (*Shift* + *Z*), to about a factor of 5.
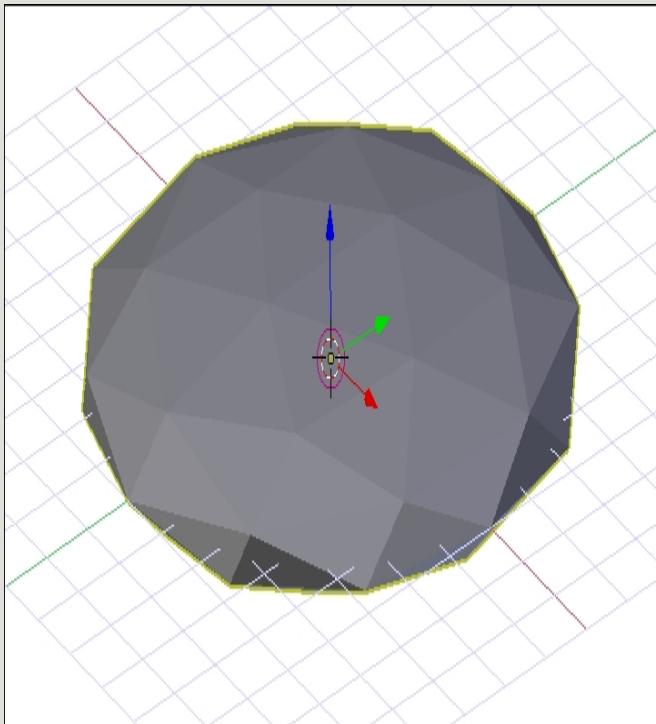
3. When the disk is the right shape, apply (*Ctrl* + *A*) **Rotation & Scale**.
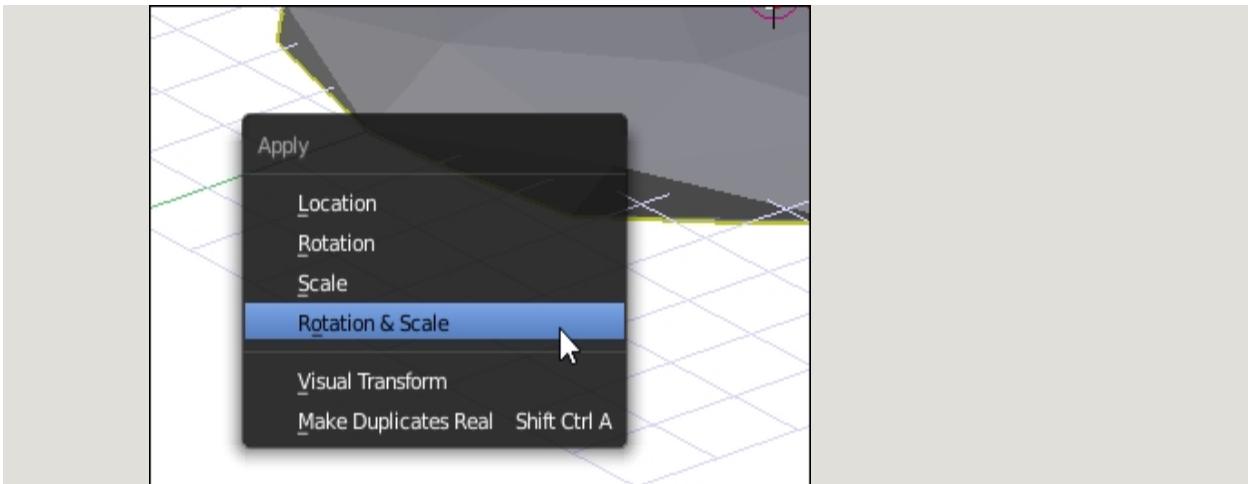
TIP

Applying rotation and scale doesn't have any visible effect on the model, but it will affect other things, particularly those related to the sculpt tools. Before applying rotation and scale, the object you see was actually another object modified with rotation and scale and could be undone. After applying it, the object you see is what the object actually is. It bakes in the changes. This is important because some actions, such as sculpting and some modifiers, are actually applied to the base object.

Setting up sculpt

When entering sculpt mode for the first time, it's worth knowing the settings available to you and becoming acquainted with a few commonly changed defaults:

1. In the menu bar at the bottom of the **3D View** panel, click on the **Object Mode** dropdown and select **Sculpt Mode**.
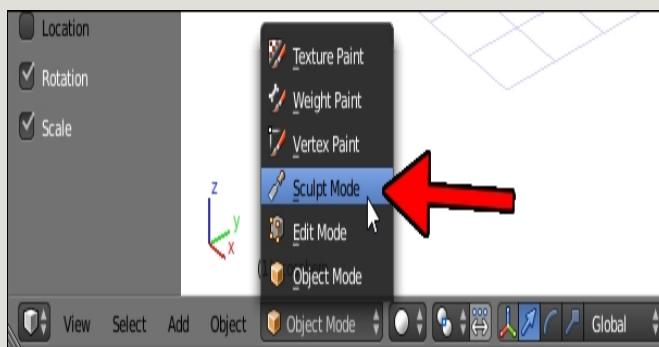
2. Once you enter **Sculpt Mode**, the tool shelf to the left of the **3D View** panel will change. In the **Tools** tab, there are a number of sections. The most common settings are in the **Brush**, **Symmetry/Lock**, and **Dyntopo** sections. Like most things in Blender, these can be customized and rearranged, so it might be desirable to change the order of these sections to move them to the top.

The **Dyntopo**, or **dynamic topology**, section has settings related to a relatively new feature for sculpting in Blender. Dynamic topology will increase or decrease the complexity of the model in whatever local area needs it in order to make the details we want to add. In other words, if we want a lot of detail on the top of the model where we'll be doing all our work, but not the bottom, which is mostly going to be ignored, we don't have to worry about making the whole model more or less complex. Dynamic topology will handle all that for us. But it needs to be turned on.

3. Open the **Dyntopo** section and click on the **Enable Dyntopo** button.

4. Change the **Detail Size** setting to 5.00 px.

The **Detail Size** value is how big the polygons will be made when sculpting. Notice that this setting is in px or pixels. This is a setting based on the view, not any actual measurements. If the view is zoomed in closer to a model, there will more detail added. When you zoom out, the detail may be reduced, eliminating some smaller details.

> **TIP**
>
> If you find that your computer isn't able to handle as much detail as is being added to the models, you can adjust the **Detail Size** setting to a bigger number. If you decide you want more detail, you can set the **Detail Size** setting to a smaller number.

5. In the **Symmetry/Lock** setting, make sure the button marked **X** under **Mirror** is highlighted.
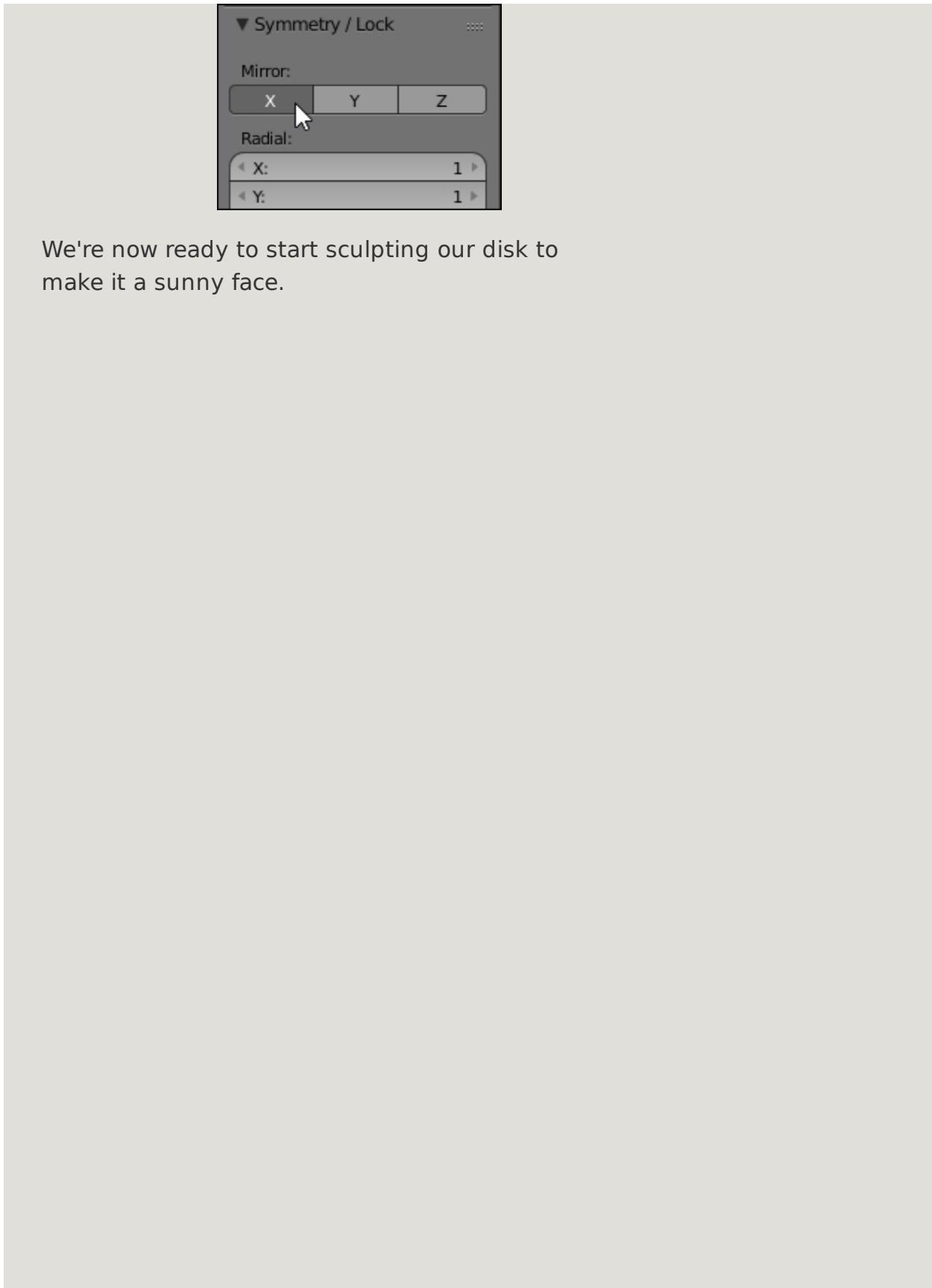
We're now ready to start sculpting our disk to make it a sunny face.

Drawing the face

In this section, we will make the face for our puzzle. Under the **Brush** tool, there is a whole palette of tools to use:
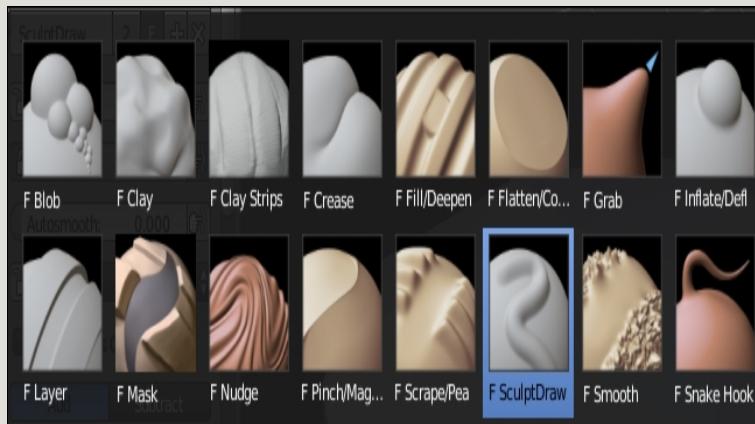
The default brush, **F SculptDraw**, is the most common and is already selected, so we'll start with it. Switch to the **Top** view (Numpad *7*) and follow these steps:

1. From the middle, nearer to the bottom, click and drag in a curve to make a smile:
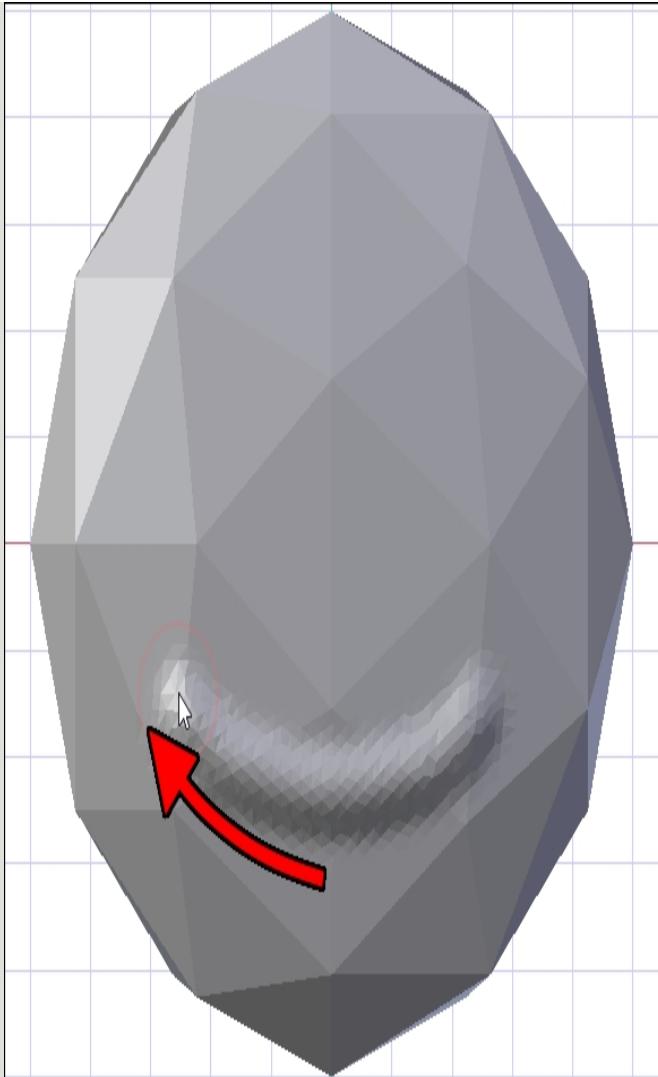
2. Repeat the action a little lower down to finish the lips:
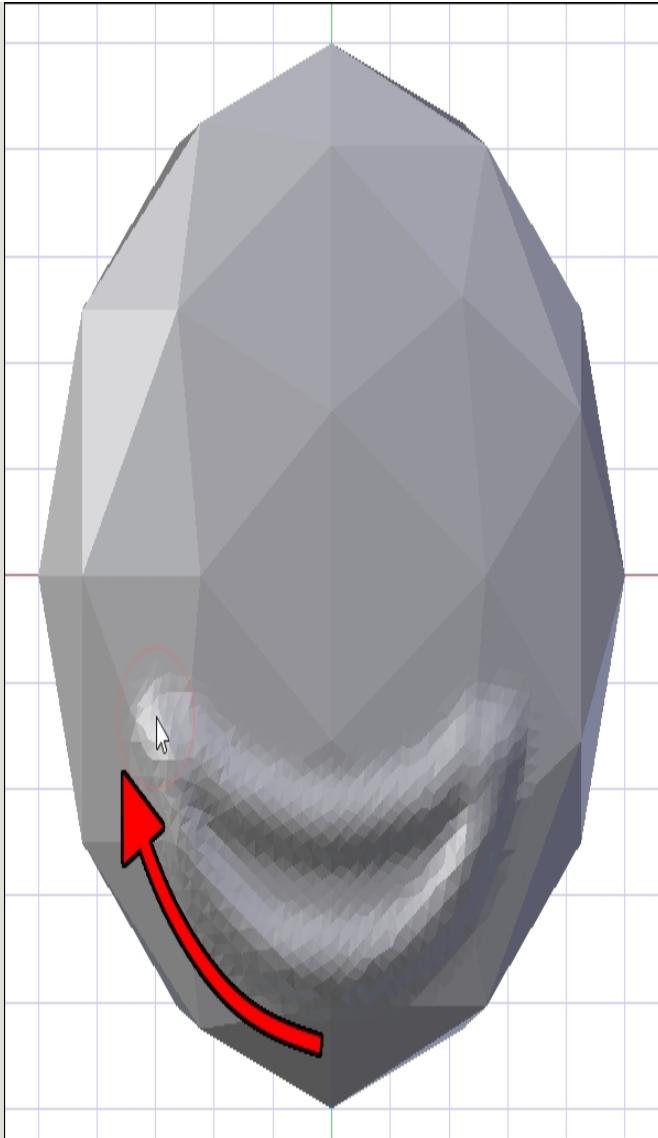
Dynamic topology will continue to add detail where we need it as we sculpt:

1. In the **Tool Box**, increase the **Radius** setting under the **SculptDraw** tool to about 70 px:
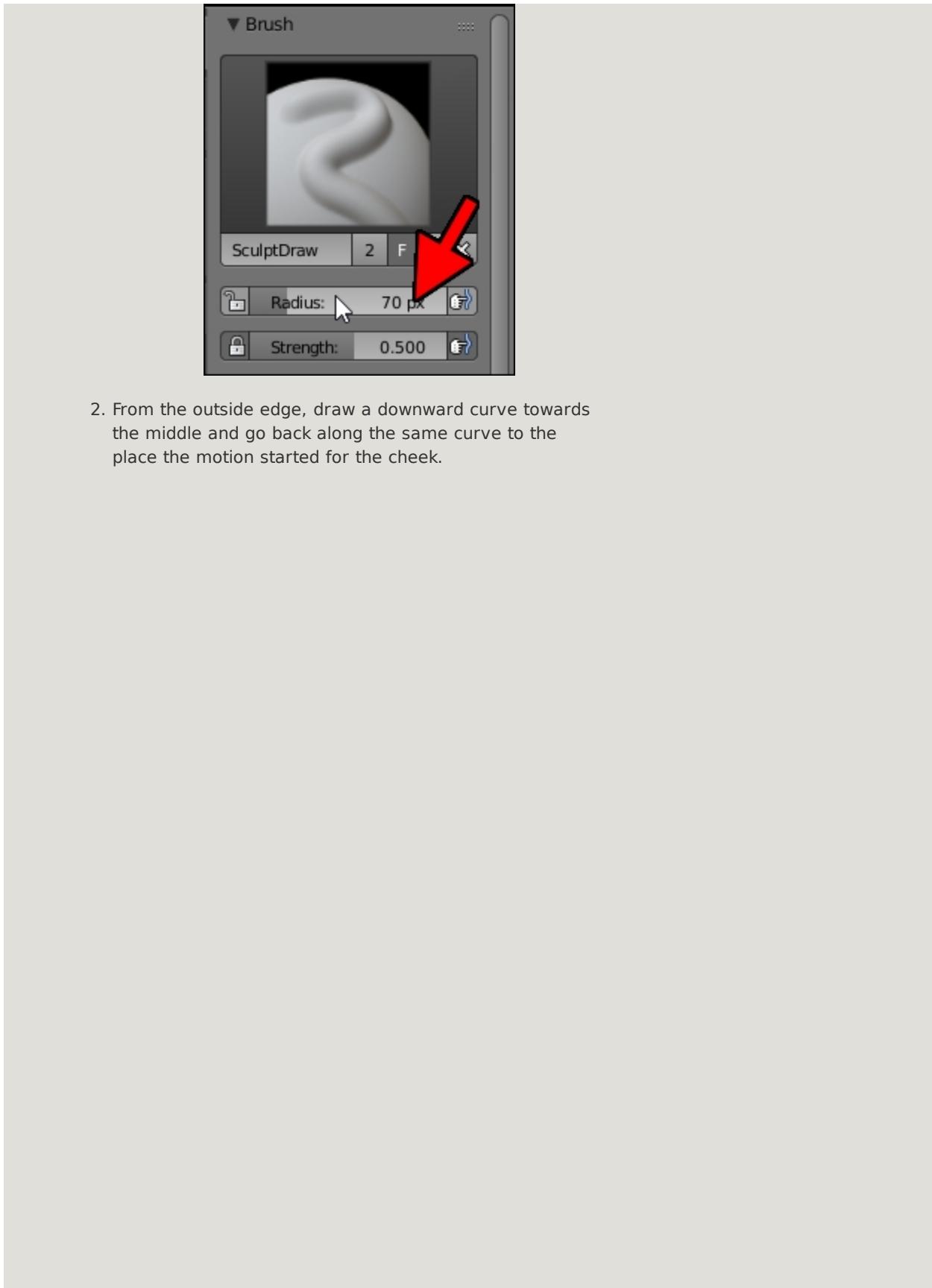
2. From the outside edge, draw a downward curve towards the middle and go back along the same curve to the place the motion started for the cheek.
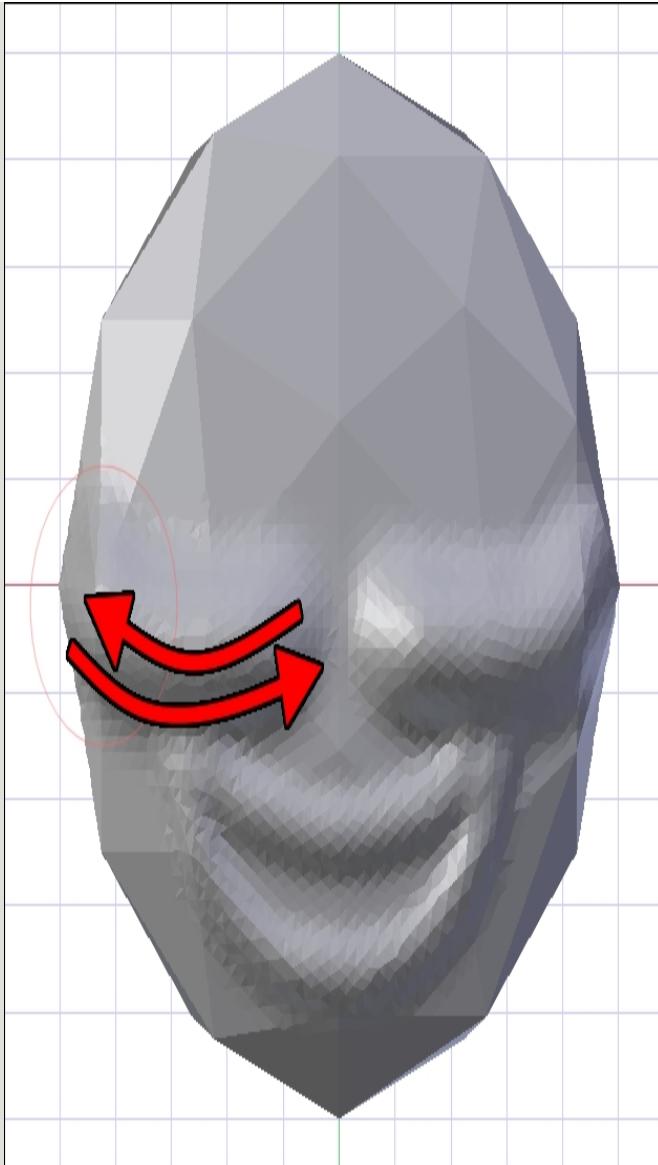
By repeating the motion back and forth, you can make the line taller than the lines for the lips if viewed from a different angle.

3. From the outside edge again, a little higher, draw an upward curve toward the middle for the eyebrows.
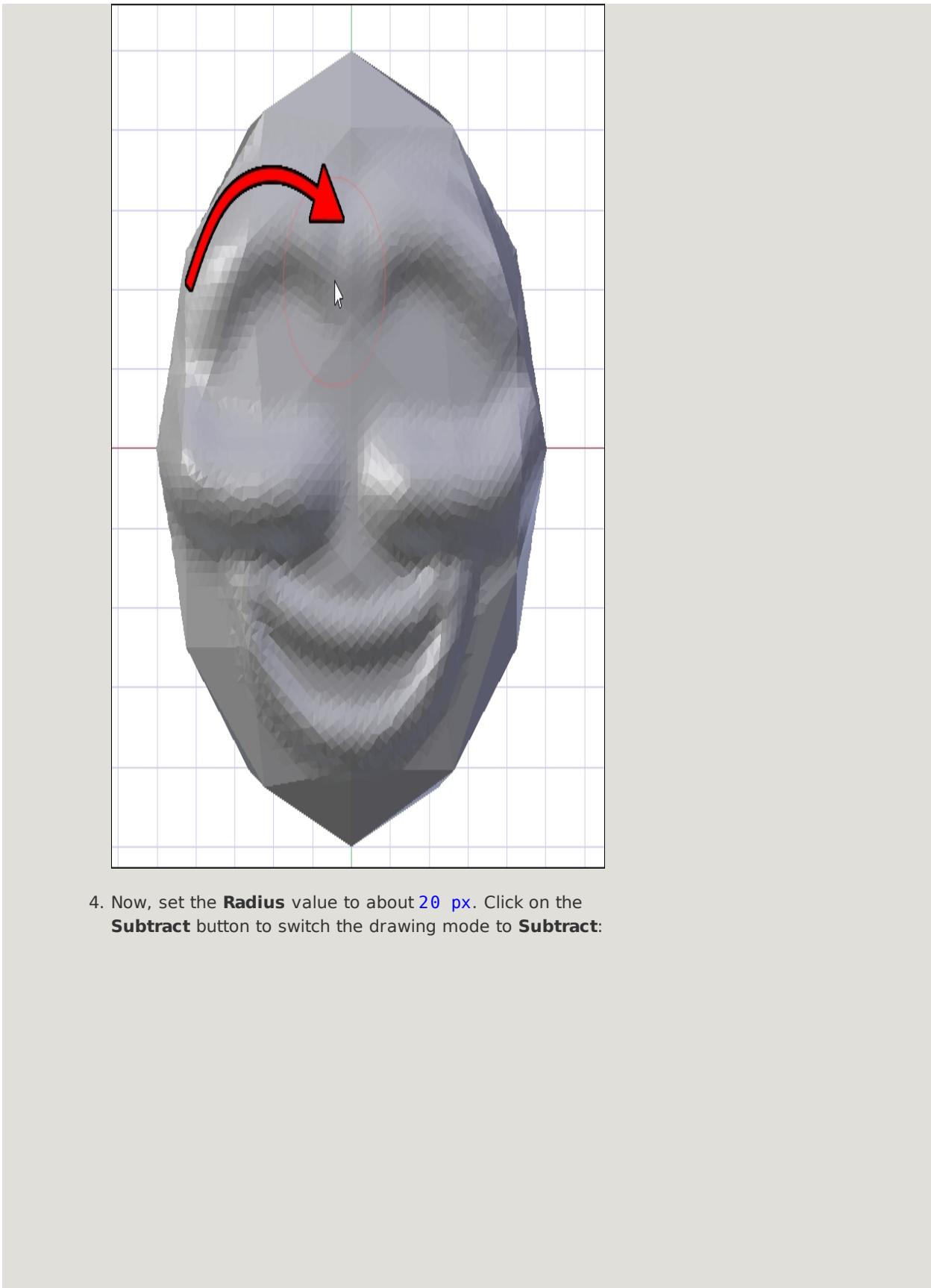
4. Now, set the **Radius** value to about `20 px`. Click on the **Subtract** button to switch the drawing mode to **Subtract**:
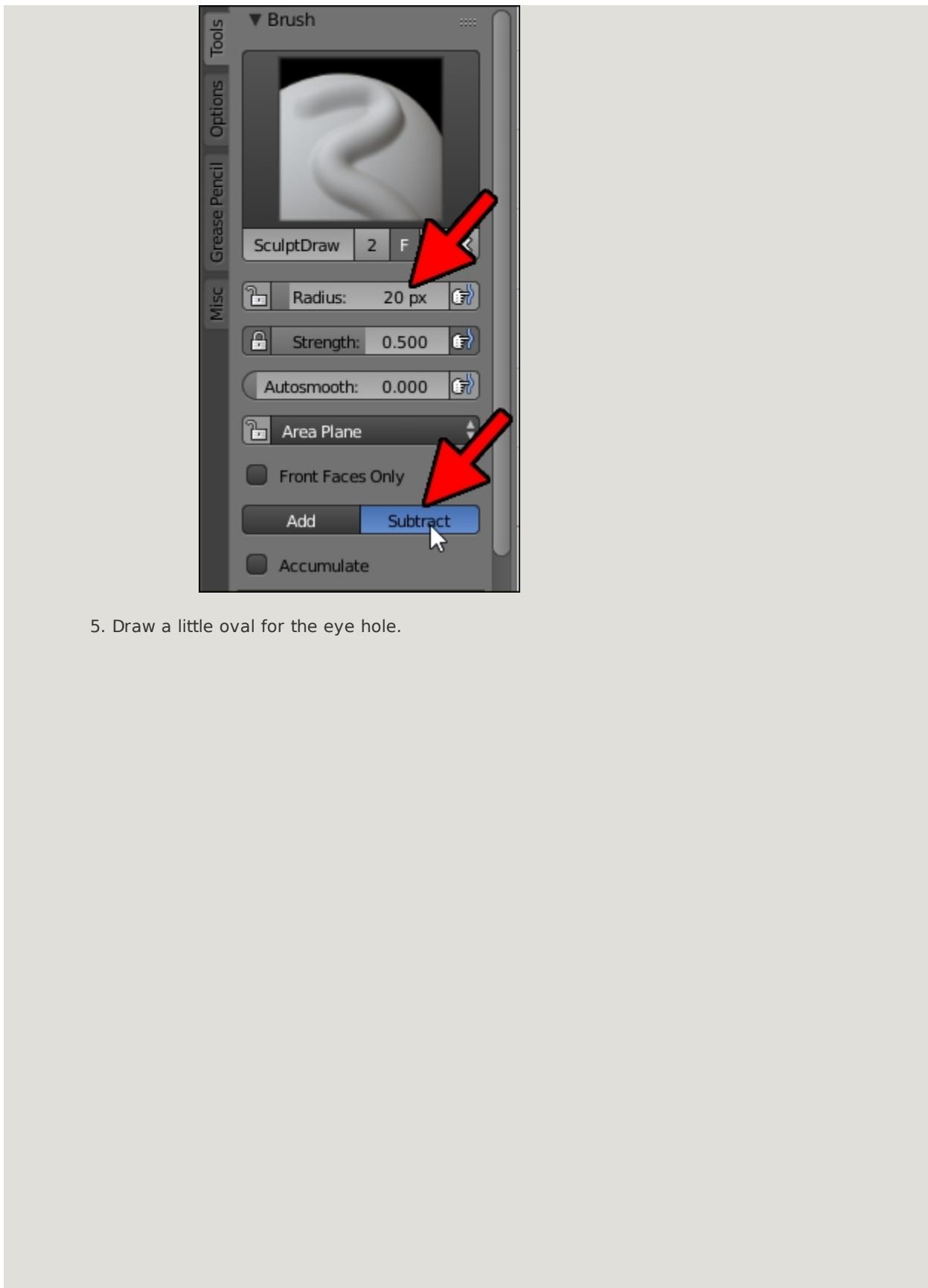
5. Draw a little oval for the eye hole.
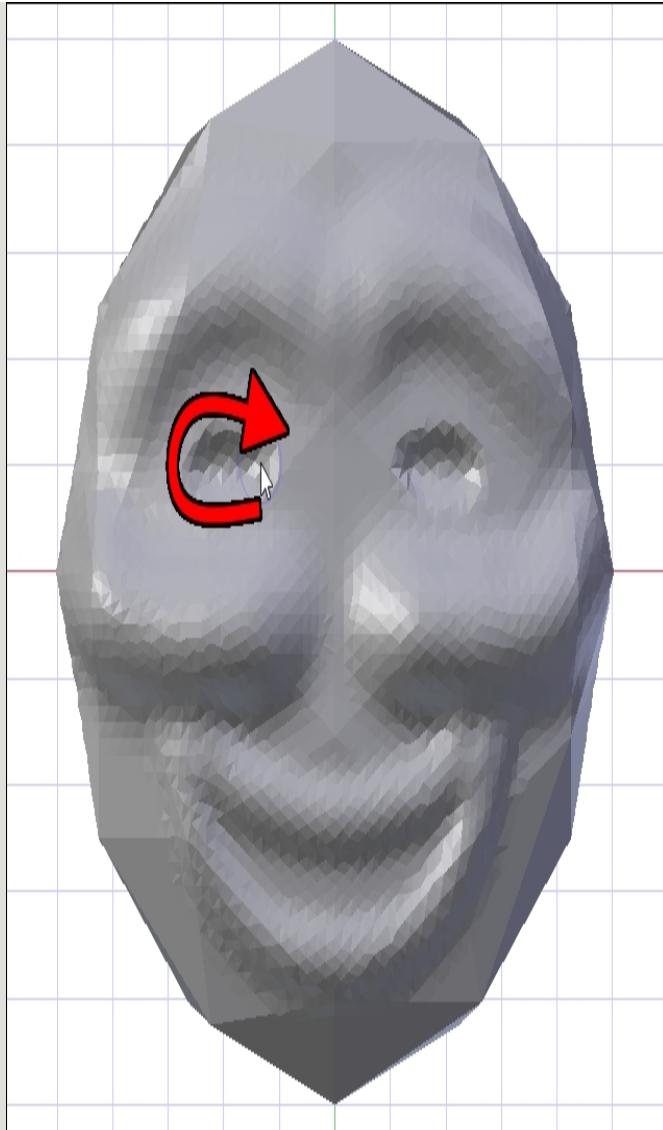
Smoothing the edges

At this point, the rough shape of the face is starting to develop, but now we need to smooth some things out. For this, we have the smoothening tool. Follow these steps:

1. Click on the **Brush** tool icon to pull up the sculpt tools.
2. Now, click on the **F Smooth** tool icon.
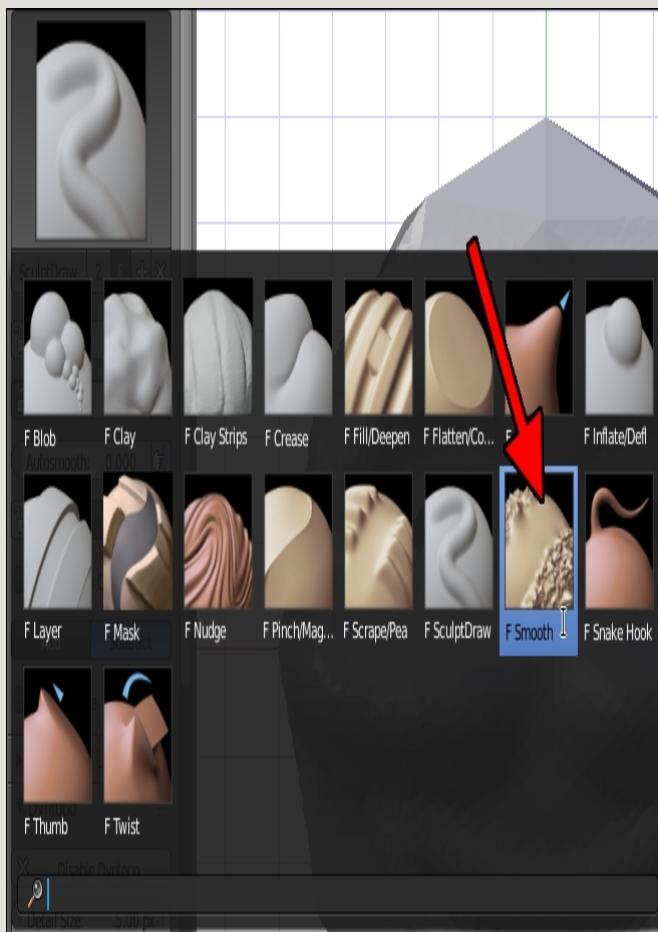
3. Draw back and forth along the curve above the lip to smooth out the upper lip.
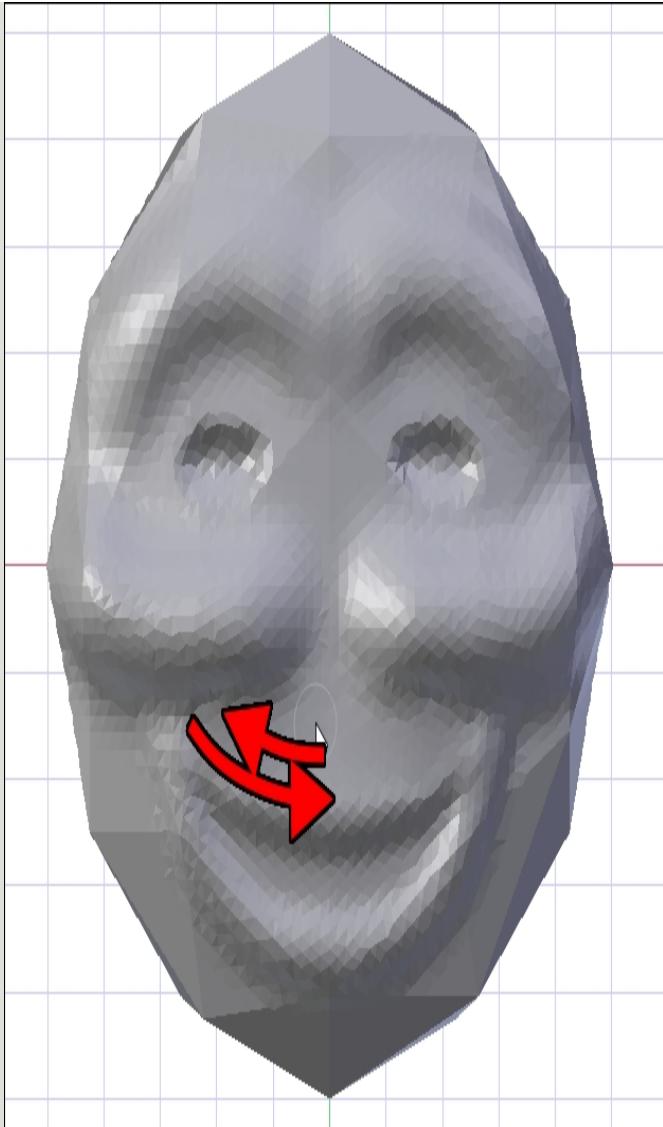
4. Draw back and forth along the curve under the lip to smooth out the chin.
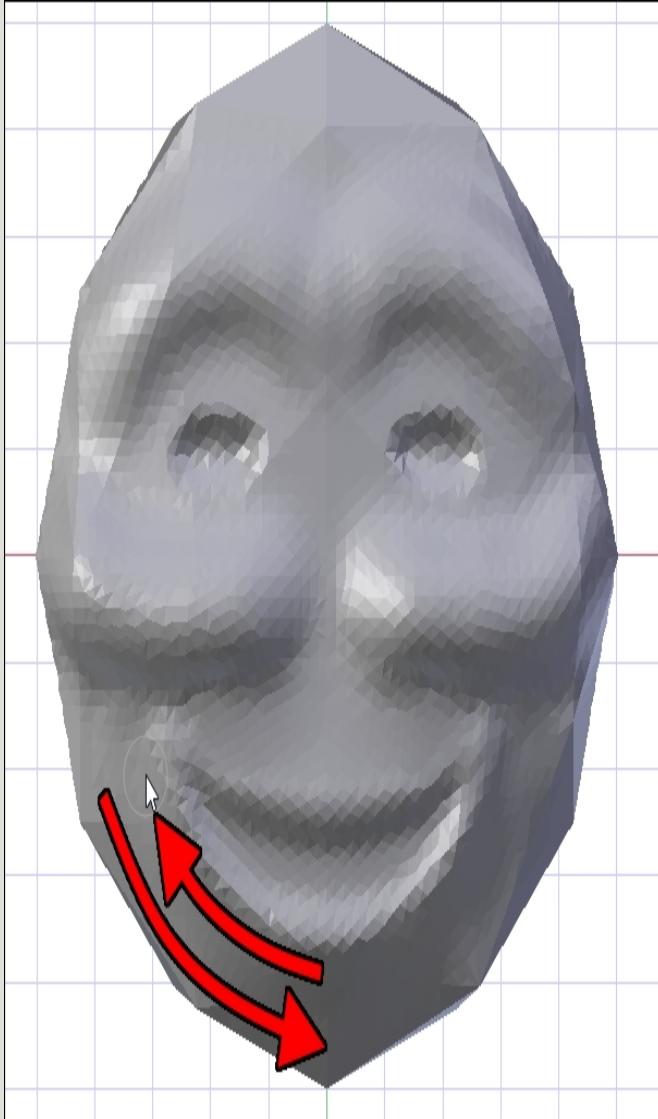
As with most things in Blender, it pays to know the keyboard shortcuts for this process. With sculpting, the easiest one is this interactive way to change the radius:

1. Hover the mouse pointer over the model and press *F* on the keyboard.
2. Move the mouse until the circle and brush shown are a little bigger than the space between the eye and the middle of the cheek.
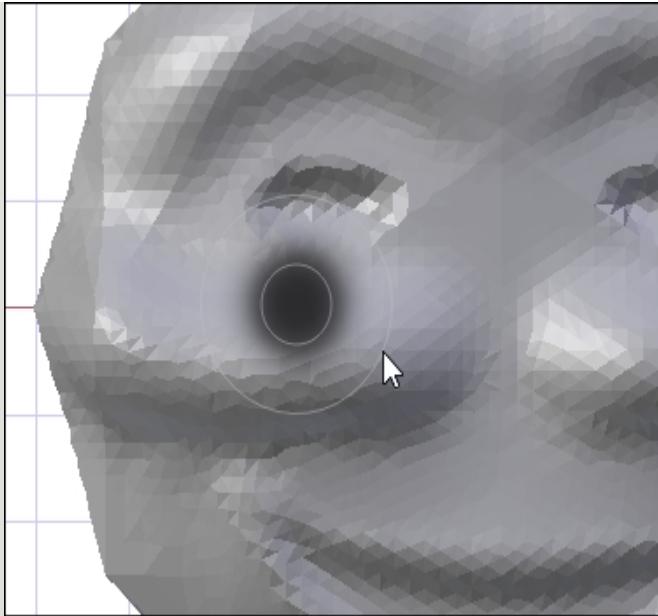
3. Click to finish the action.
4. Draw back and forth along the top of the cheeks to smooth them.
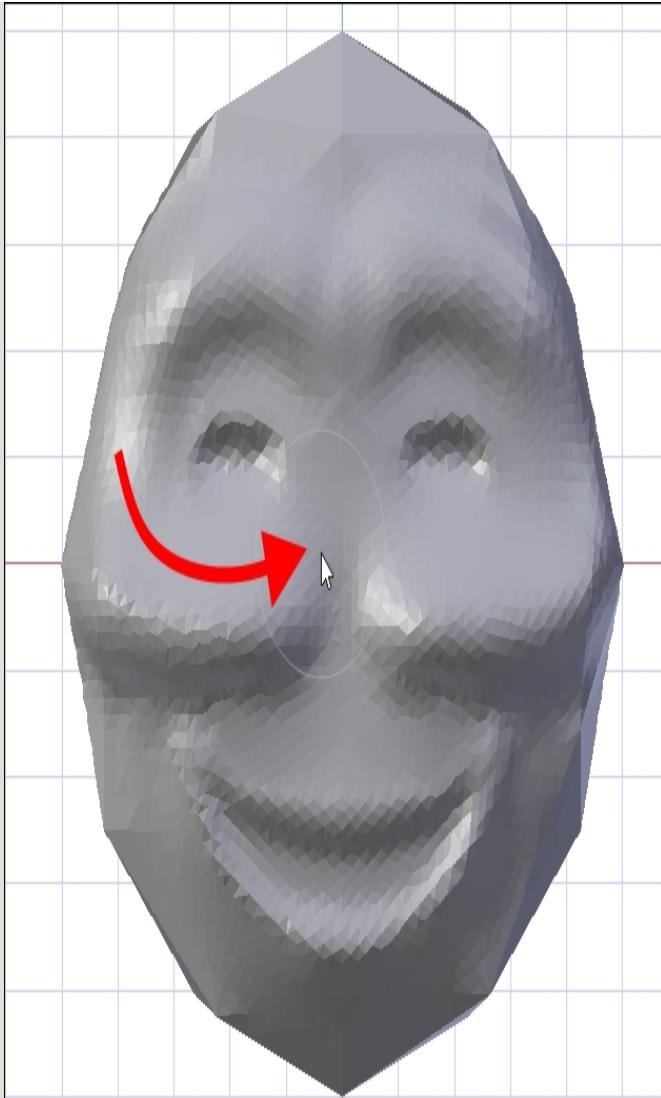
Adding the nose and eyes

To add the details of the nose and eyes, we'll use the **Inflate** tool. We'll also explore a few more keyboard shortcuts to simplify the process. The steps are as follows:

1. On the keyboard, type *I* to switch to the **Inflate** tool.
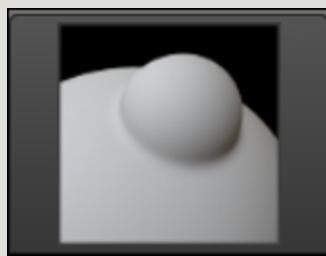
2. Use the *F* key to size the brush for drawing the nose.
3. Size the brush and click to complete the action:
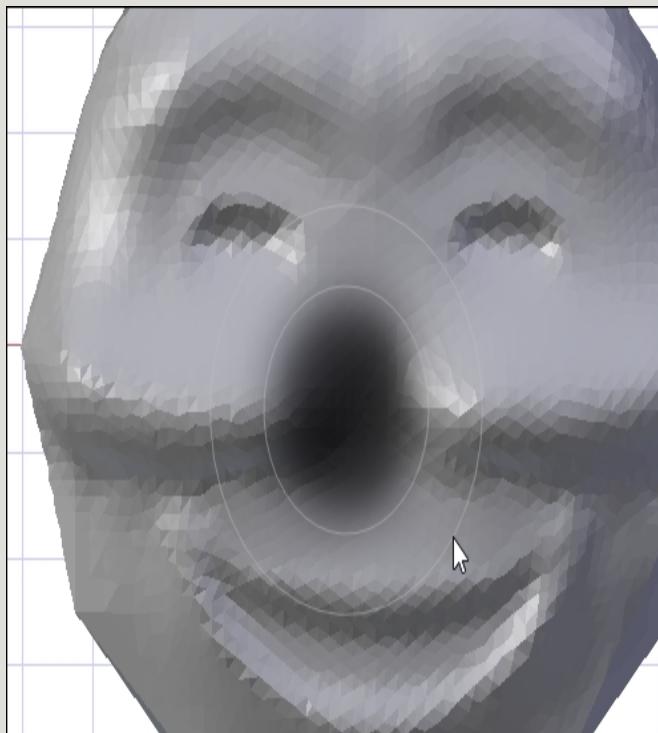
TIP

Strength is another setting that can effect sculpting tools. You may have noticed it under the **Radius** setting. Setting **Strength** closer to 1 makes the effect of the tool stronger, while setting it closer to 0 makes it weaker, with 0.5 being a happy medium setting between them. For drawing the nose, a dramatic change by setting the **Strength** to 1 will save time and effort and, fortunately, there is a keyboard shortcut for doing that.

4. Press *Shift + F* on the keyboard to change the **Strength** setting of the tool.
5. Move the mouse pointer toward the middle of the **Radial** tool to increase the strength to 1.00.
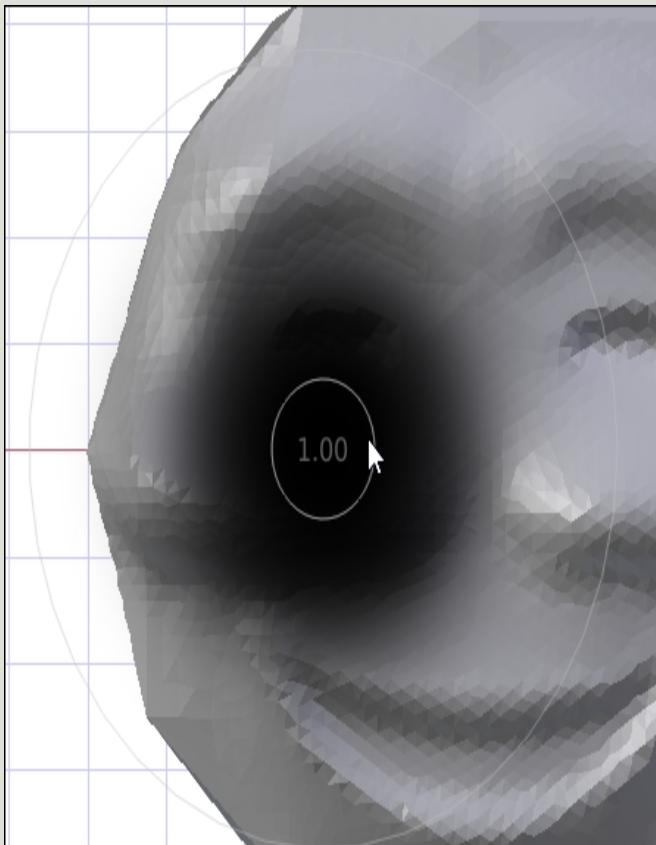
6. Click to set the strength.
7. Draw little circles around the nose area to build up the nose.

8. Hold down *Shift* while drawing in a line from the nose to where the eyebrows meet to quickly smooth out the bridge of the nose.
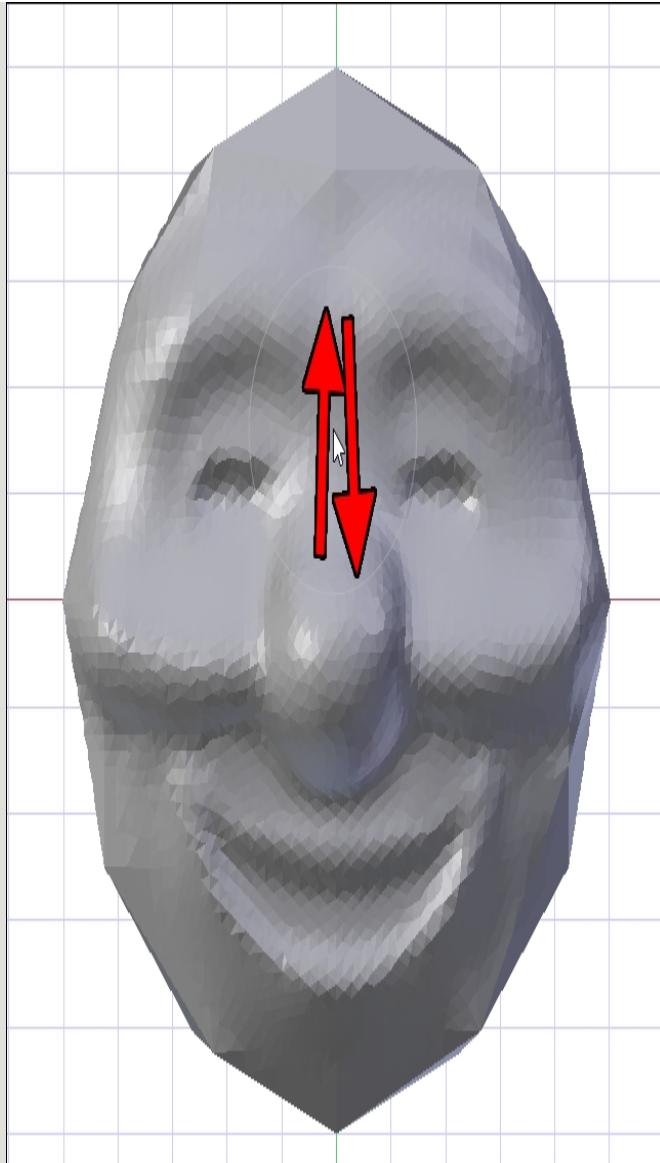
9. Adjust the view to be looking up at the nose from below.
10. Adjust the **Brush** size (*F*) to be about the size of a nostril.

User Ortho
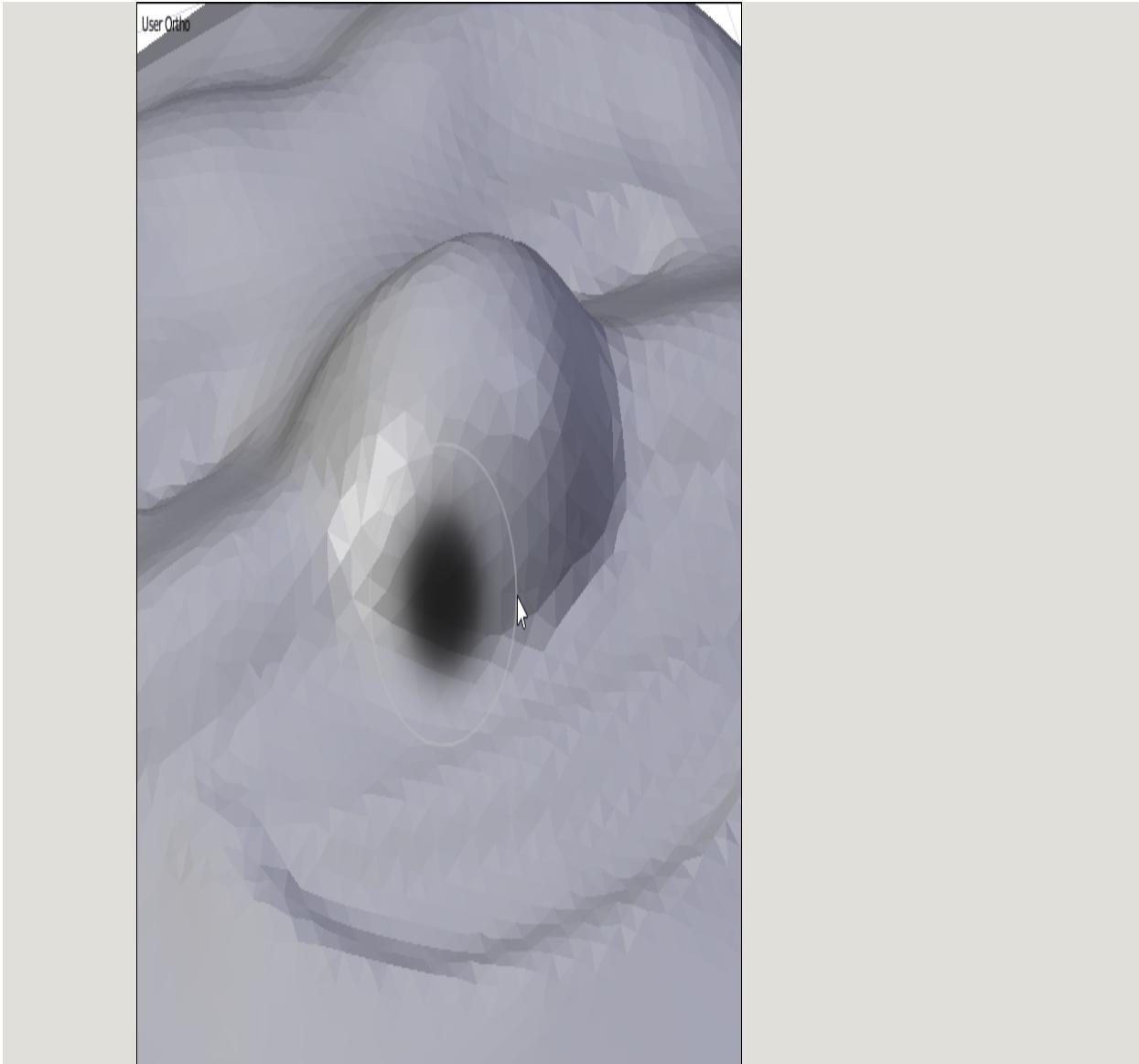
TIP

Holding down *Ctrl* while using a sculpting tool toggles the effect, switching add to subtract or inflate to deflate.

11. Hold down *Ctrl* while drawing in little circles to deflate the nose holes into the nose.
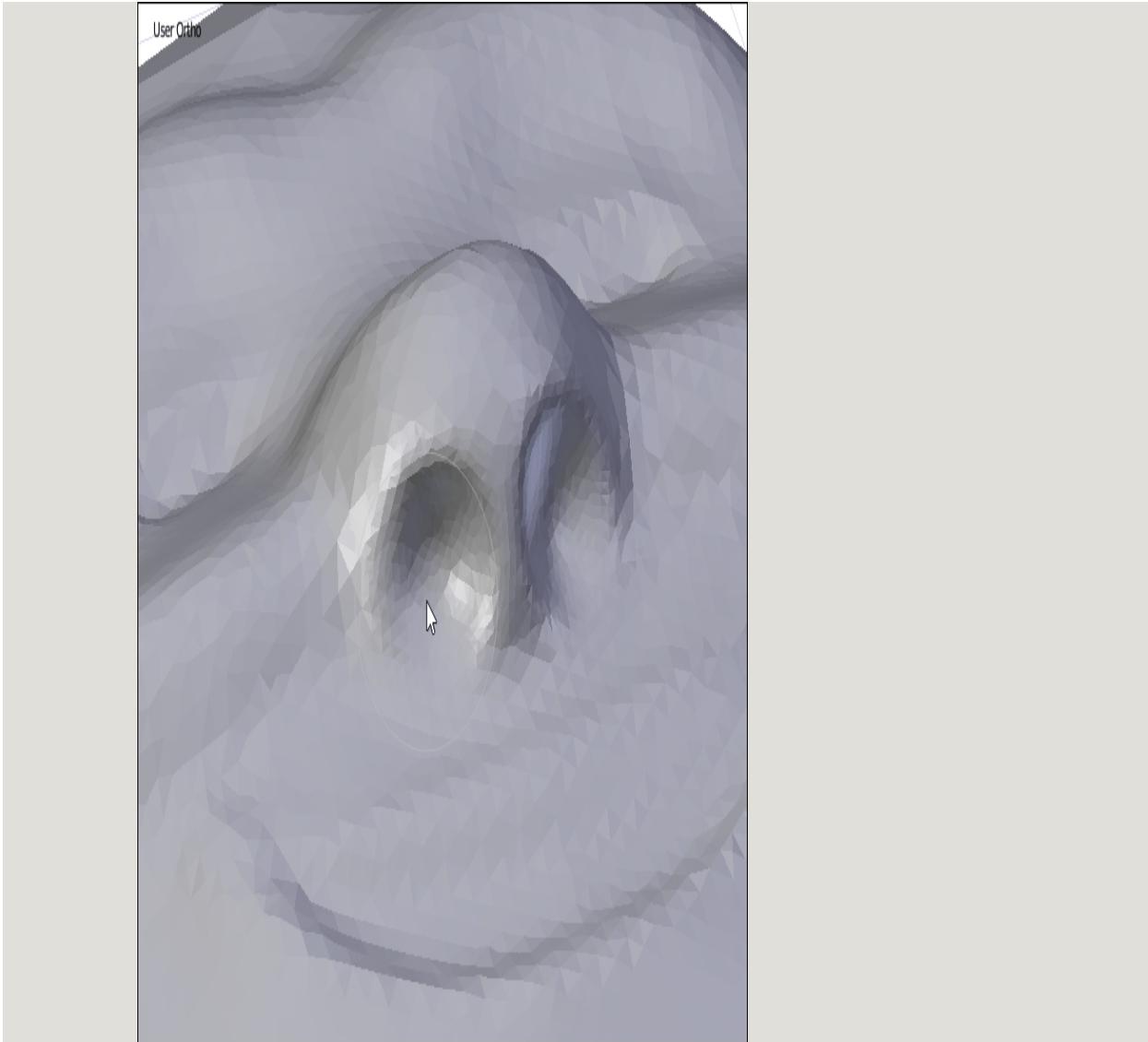
12. View from the top (Numpad *7*) and zoom in the view to one of the eye holes.

13. Resize the **Brush** (*F*) to be a little smaller than the eye hole.
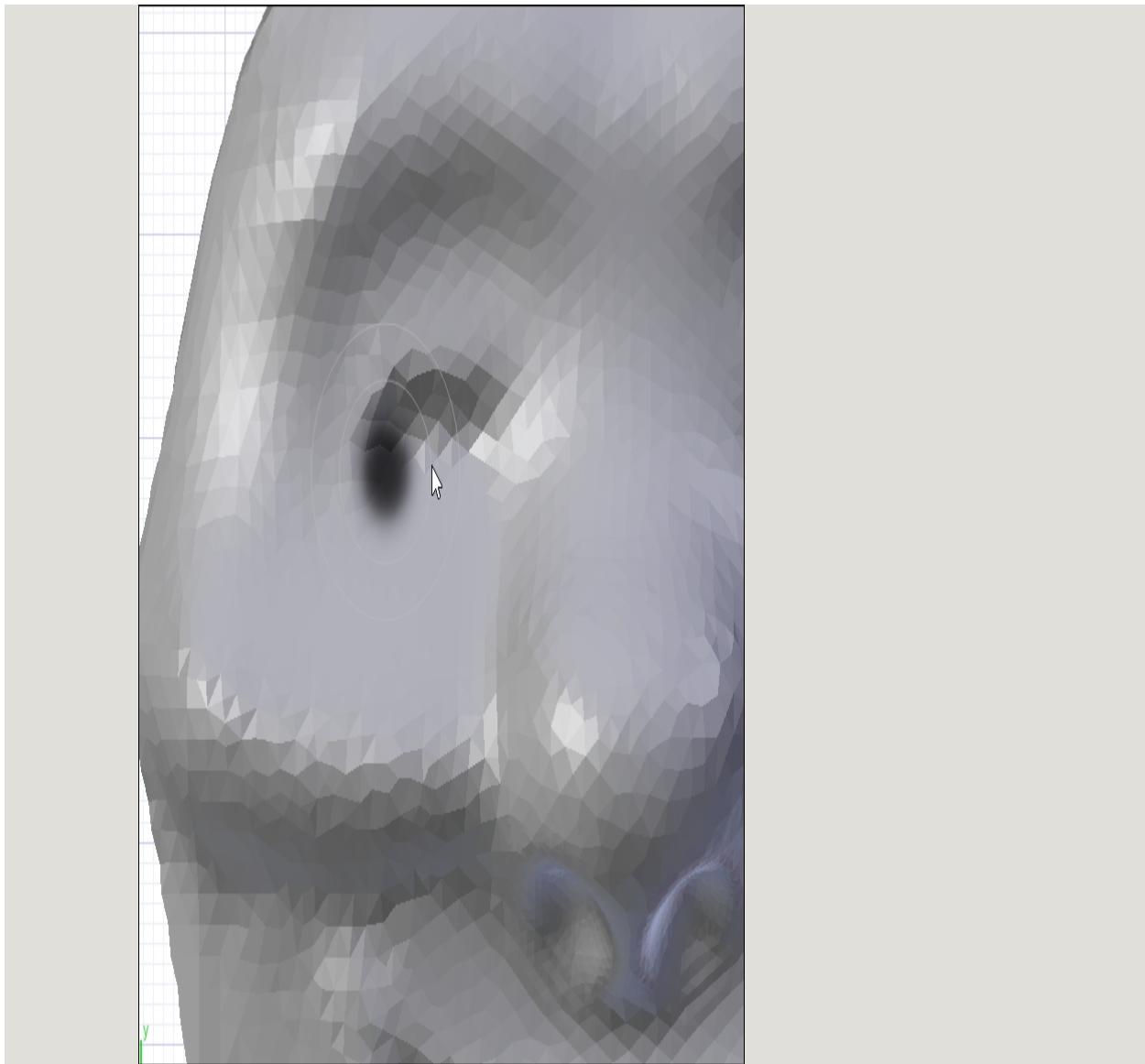
14. Draw little circles in the eye hole to build up an eye.
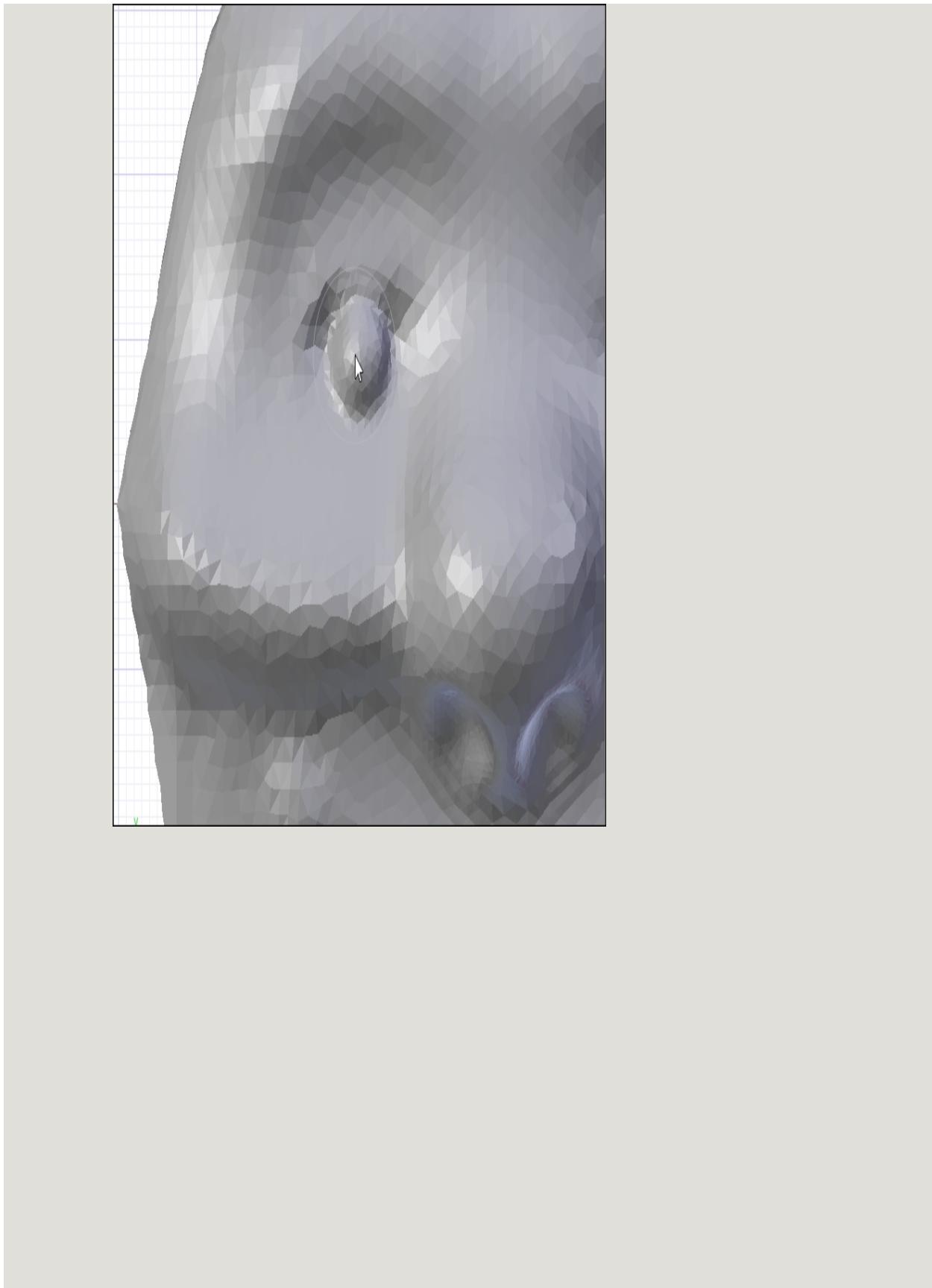
Pulling out the rays

We'll add more details to the face in a minute. First, let's finish out the main shape of the sun by pulling out the rays of the sun. The goal here is to make something curvy and artistic to look at, so for these parts, symmetry needs to be turned off first. Follow these steps:

1. On the left-hand side of the **3D View**, navigate to the the **Symmetry/Lock** part of the **Tools** tab in the **Tool Shelf** and click on the **X** button under **Mirror** to toggle X-axis symmetry off.

2. Switch to the **Snake Hook** brush (*K*).

3. Adjust the view and change the **Radius** (*F*) so that the brush is about big enough to cover about an eighth of the edge of the face.
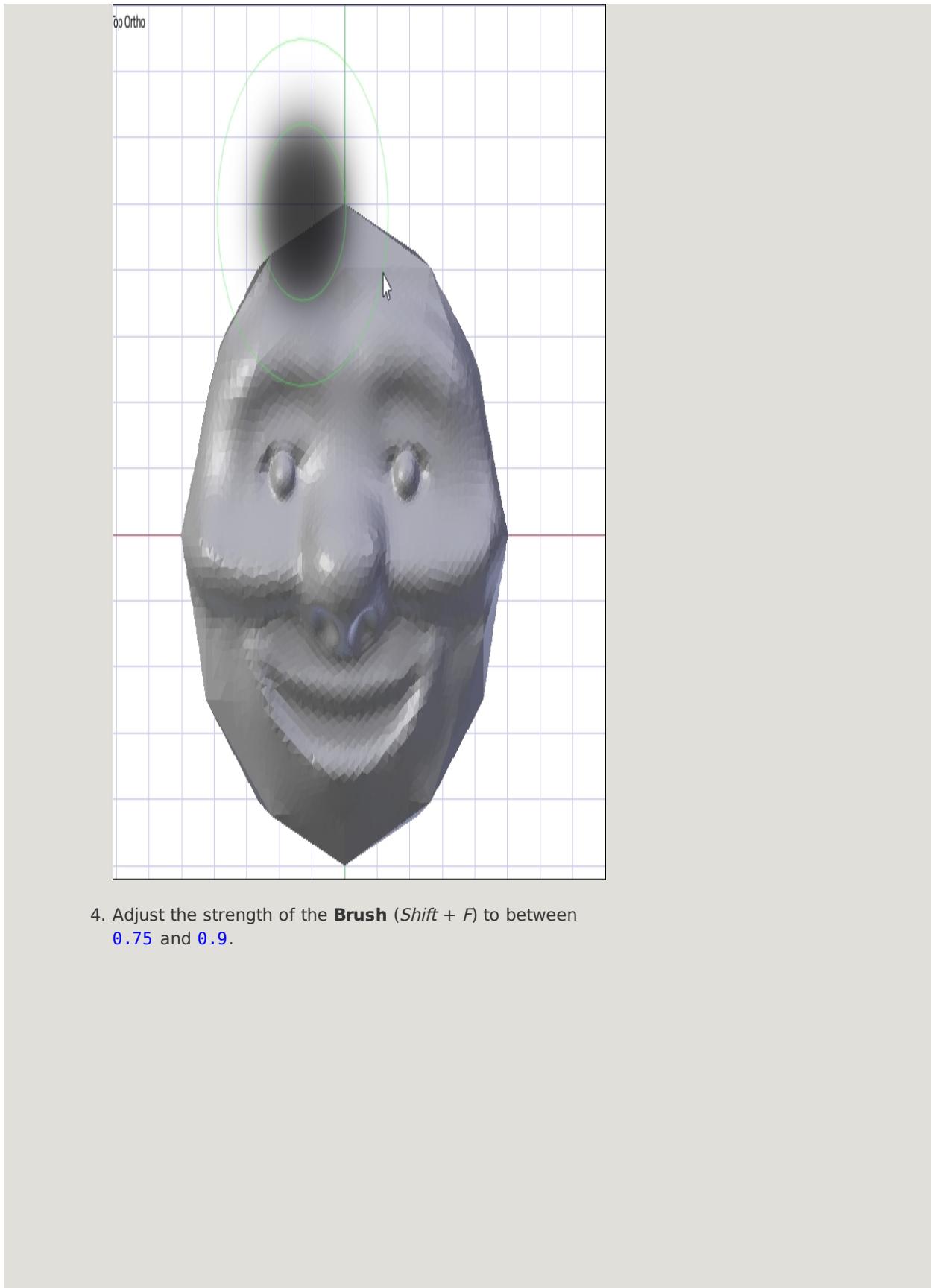
4. Adjust the strength of the **Brush** (*Shift* + *F*) to between 0.75 and 0.9.
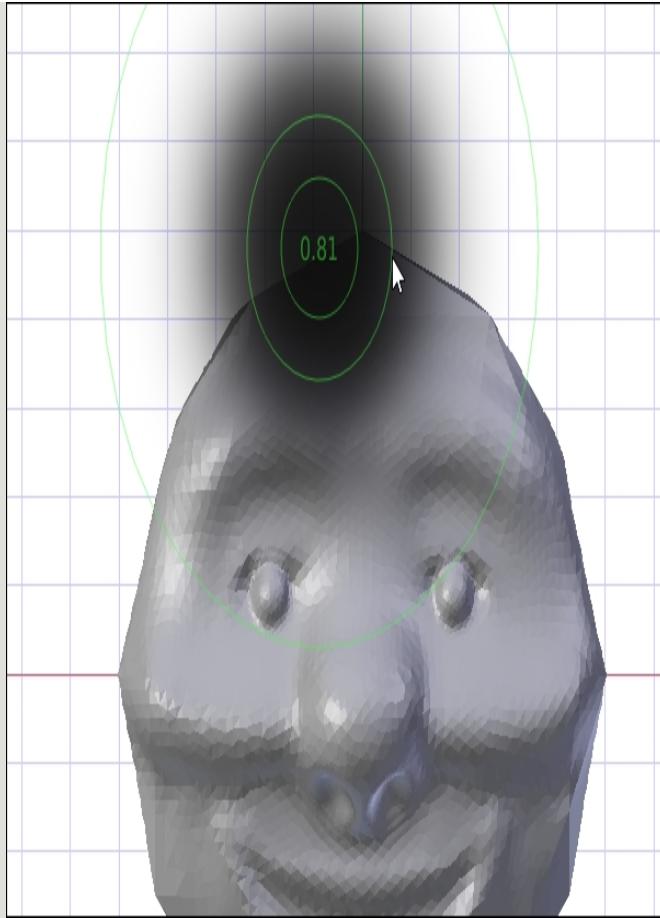
5. From the top of the head, being sure the center of the brush catches at least a little bit of the edge of the head, start drawing while moving the mouse in a curve away from the body.

Top Ortho
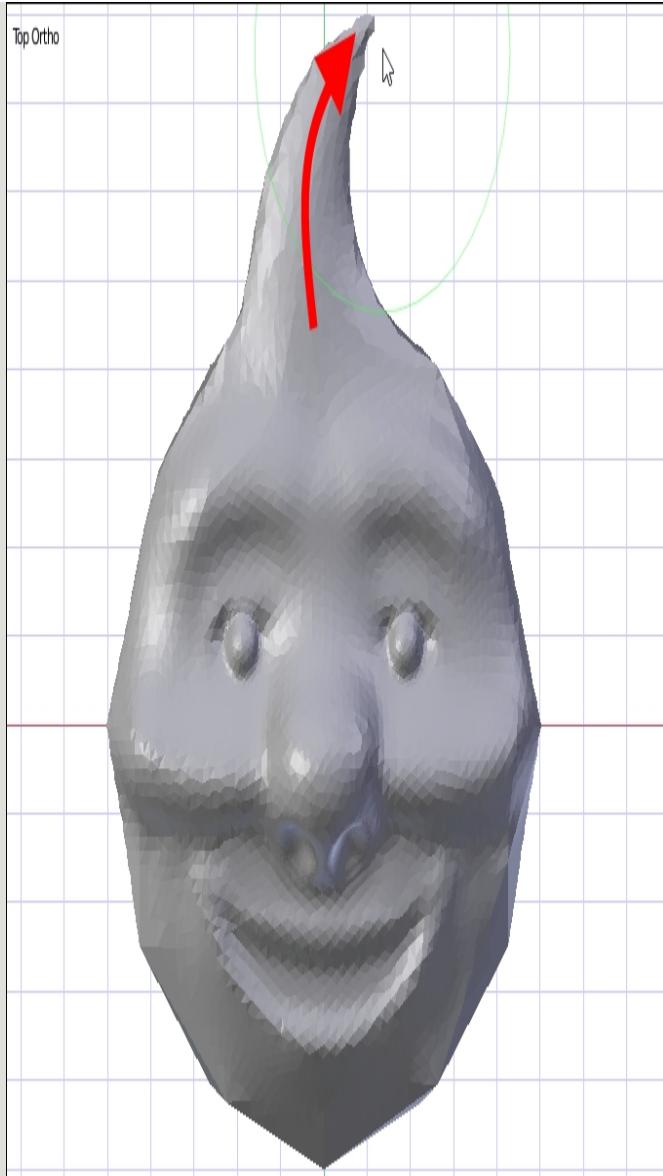
If the rays are being pulled out too far, undo the action, adjust the strength, and try again. If the brush is grabbing too much of the face, undo the action, adjust the radius, and try again.

6. Continue around the edge of the face, pulling out rays with the **Snake Hook** tool.

The **Snake Hook** tool is a useful tool because it quickly creates new areas that can be further sculpted. It can be used to make arms, tentacles, horns, or other shapes. Usually, it's preferable to generate a basic shape that is approximately shaped like the object to sculpt, but with **Snake Hook**, this doesn't necessarily have to be the case.

Sculpting is a more organic process than other modeling, so as you become comfortable with

the tools, be sure to experiment with them and make the design your own.

Sharpening the details

At this point, the general shape of the face is set, but there can be more detail added. Keep in mind that with 3D printing, there is such a thing as too much detail, since some of it won't come through. Then again, details tend to be the most impressive part, so spending some time sharpening some edges may be of benefit. Here's how:

1. In the **Tool Shelf**, under **Symmetry/Lock**, click on the **X** button under **Mirror** so that it is highlighted in order to toggle mirroring back on.
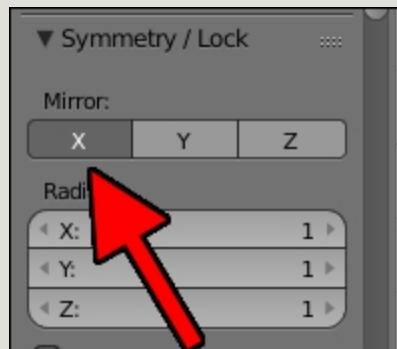
2. Switch to the **Crease** tool (*Shift + C*).
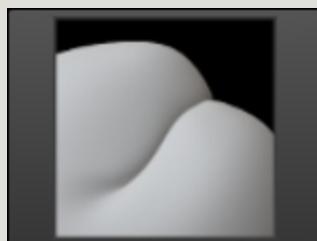
3. Make sure the **Add** button in the **Tool Shelf** is highlighted. If not, click on it so it is highlighted.

4. Adjust the view and change the **Radius** (*F*) until the brush is big enough to cover the bridge of the cheek.
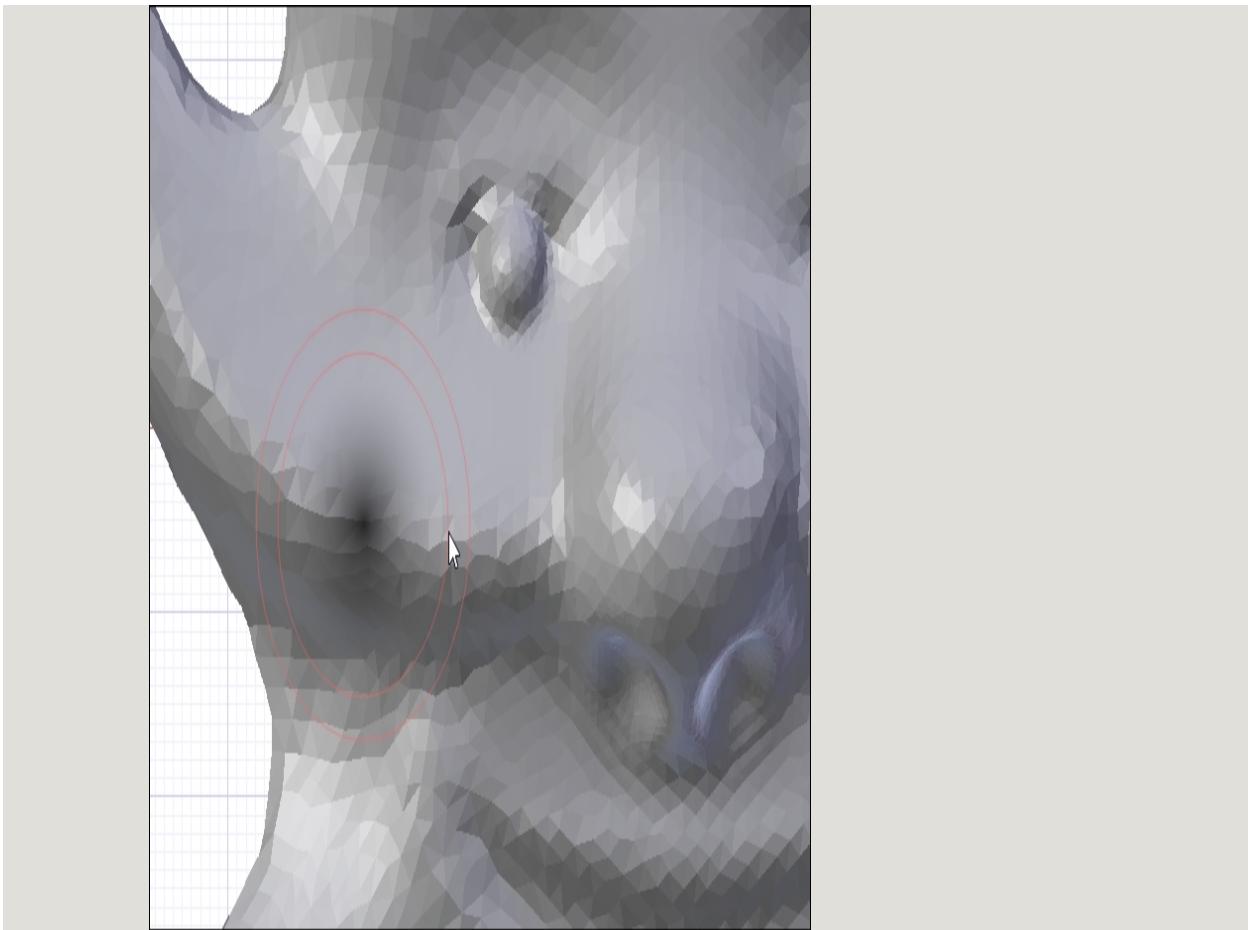
5. Draw a line following the bridge of the cheek to sharpen it a bit.
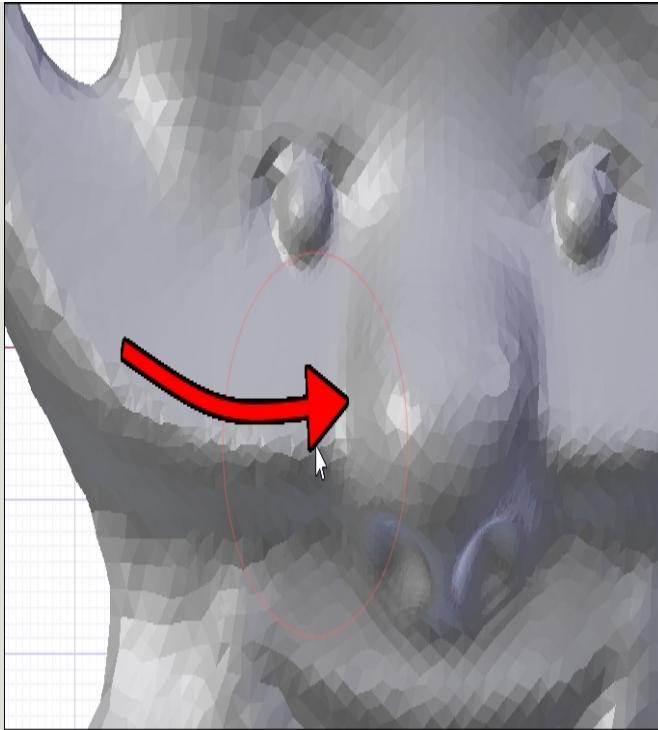
6. Draw in a line around the eyebrows and down the bridge of the nose.
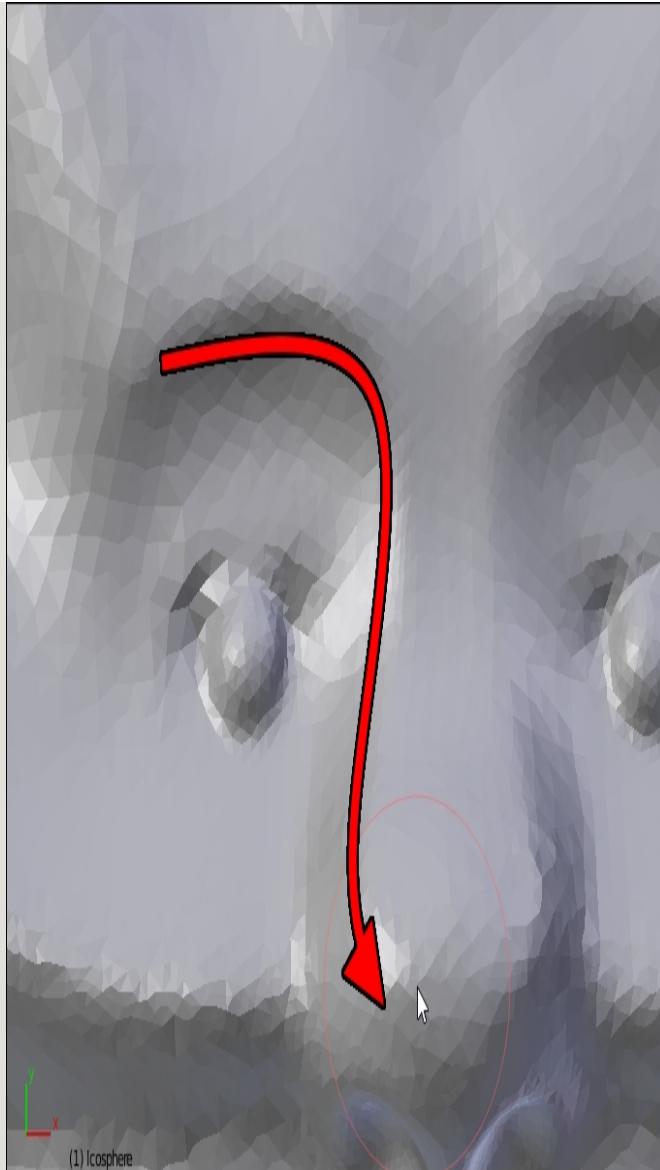
7. While holding down *Ctrl*, draw a line following the curve of the mouth.

8. Adjust the view to zoom in on one eye.
9. While holding down *Ctrl*, deepen the crease around the eyes and maybe add some smile lines.

The **Crease** tool is great for adding small details such as lines or ridges to the model and can really give your design some style. But use caution—additive creases can create undesired ridges if overused. Be sure to adjust your view to make sure that isn't happening. If it does, remember that you can undo or smooth quickly without switching brushes by holding down *Shift* while drawing at any time as well. If elements aren't where you want them, the **Grab** brush (*G*) will allow you to reposition things to a certain degree.

10. Continue to adjust the look of the model until you're satisfied with the look.

For the next steps, we'll be using the **Object Mode** tools, so after sculpting is done, switch back to **Object Mode** and click on the **Sculpt Mode** dropdown in the menu of the **3D View** and select **Object Mode**:
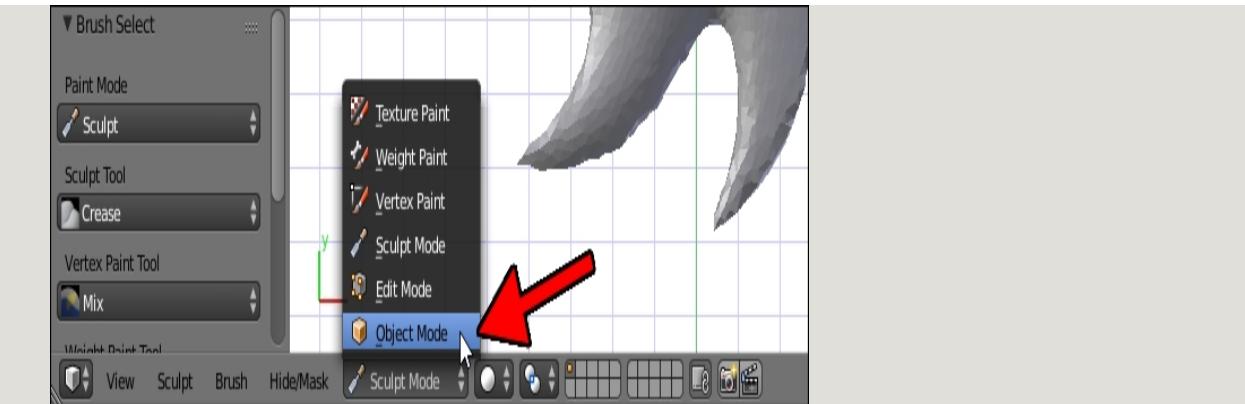
TIP

Here is a handy guide for easy reference to some of the keyboard shortcuts used in sculpt mode. For a complete set of shortcuts, be sure to check the Blender manual online.

- *F*: Change brush size
- *Shift + F*: Change brush strength
- *Shift* while drawing: Quick smooth
- *Ctrl* while drawing: Toggle brush direction.
- *X*: The **SculptDraw** brush
- *I*: The **Inflate** brush
- *Shift + C*: The **Crease** brush
- *G*: The **Grab** brush
- *K*: The **Snake Hook** brush

Summary

In this chapter, we used the sculpting tools in Blender to transform a basic shape into a complicated design. Dynamic topology managed the issues of resolution, automatically adding geometry where it was needed to create the effect we wanted. With tools such as **SculptDraw**, **Smooth**, and **Inflate**, the model quickly took shape. Hotkeys sped up the process. The **Crease** brush added some style and if it was required, the **Grab** brush gave us the power to adjust the placement of things, all with movements as natural as drawing.

In the next chapter, we'll take the model created here, combined with some more traditional modeling, and divide this model into a puzzle.

Chapter 7. Cutting a 3D Jigsaw Puzzle

The sculpting was done in the previous chapter; it's now time to cut the model into pieces suitable for 3D printing. This is accomplished with Blender's traditional object-editing tools, as opposed to the sculpting tools used in the last chapter. Blender's combination of organic and more rigid object modeling is one of the things that makes it so powerful and versatile. In this chapter, we'll prepare the model, create the puzzle space, and apply that to the model to create the individual pieces:
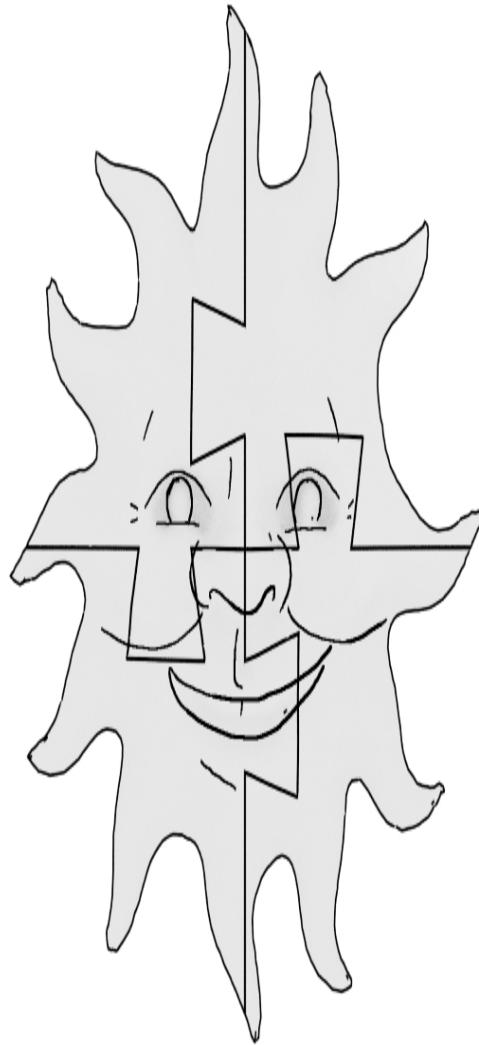
Resizing the model

While sculpting the sun face, there was very little effort made to make the model conform to a particular size. However, this is an important consideration when building the model for 3D printing because of tolerance considerations, which will be covered in more detail later.

How big should it be?

Every 3D printer has a range of print areas that it can handle. Naturally, if designing for a particular

3D printer, this should be taken into account.

For this project, the model is going to be sized so that it will be too big for any printer to print. The smallest popular 3D printer on the market has a print bed of 120 mm squared, or about 4" on each side, so that one will be what this model will be resized for.

Scaling with properties

At the end of the previous chapter, the sculpted face was left in **Object Mode**. Before continuing, make sure that the project is still in **Object Mode**, or some later steps may get confusing.
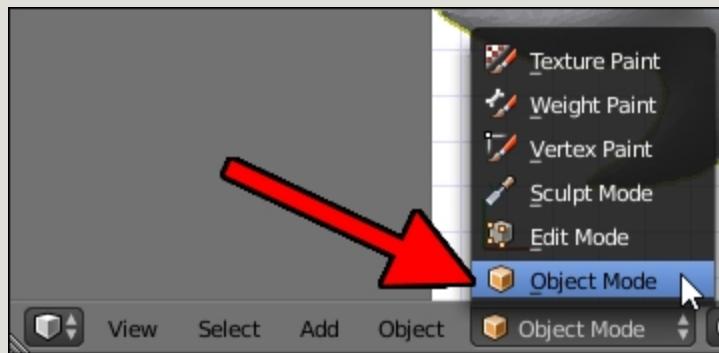
Follow these steps to scale the model:

1. In the **Properties** panel (*N*), to the right of the default view, locate the **Dimensions** pane:
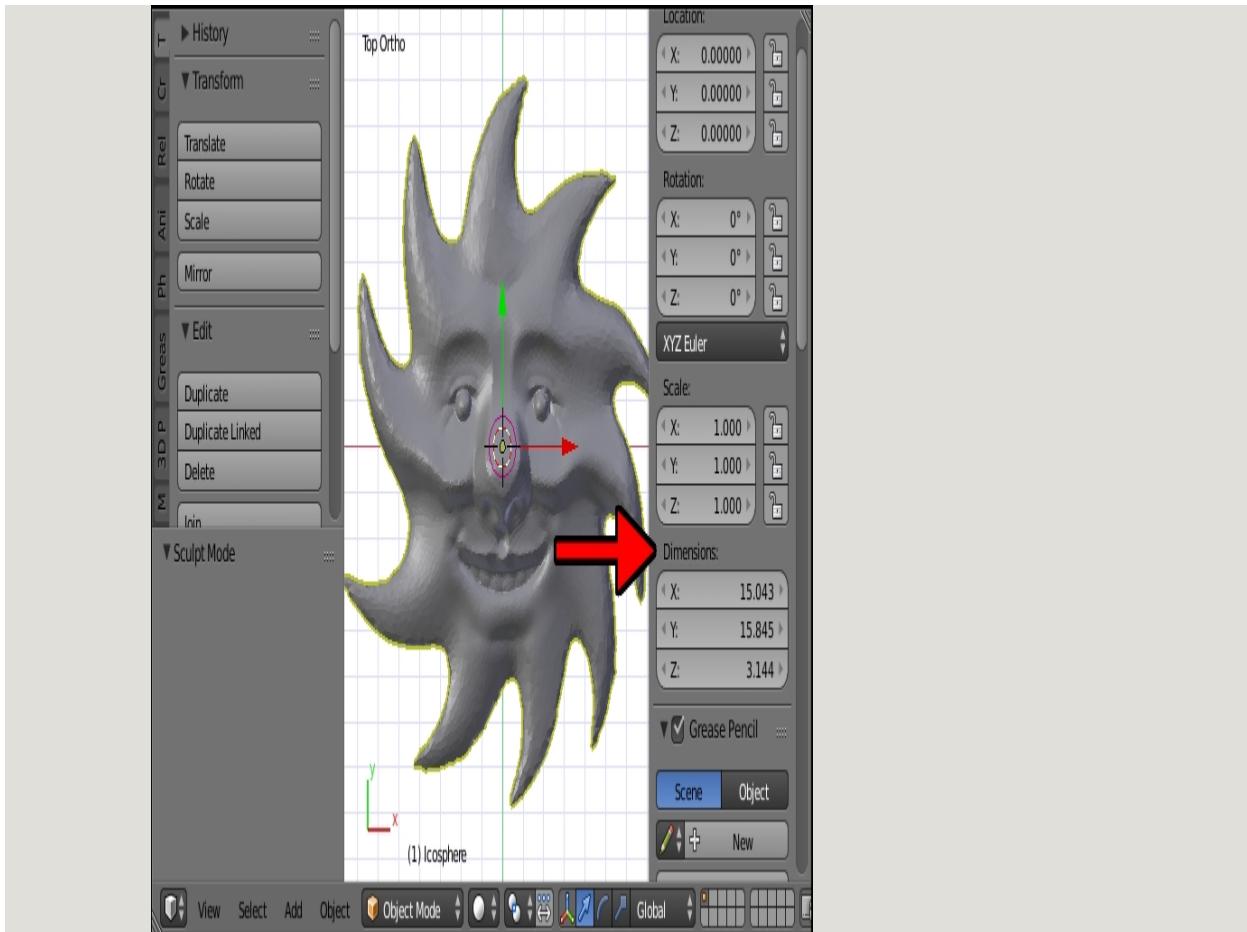
2. Change the **X** dimension to 100.

3. Now, locate the **Scale** setting, just above **Dimensions**, and make note of the **X** scale necessary to make this model 100 mm wide:
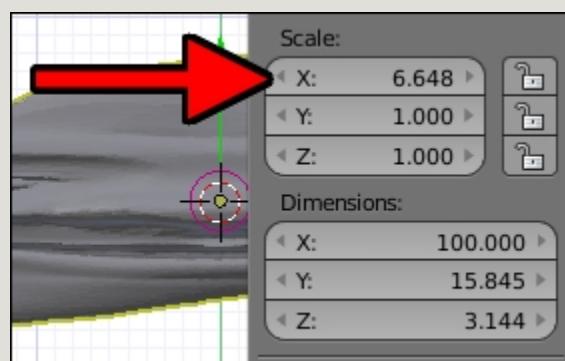
4. Round the scale value off to something easy to type (such as 6.5 in this example) and type this value in for the scale for **X**, **Y**, and **Z**:

Scale:

◄ X:	6.500 ►
◄ Y:	6.500 ►
6.5	

Dimensions:

◄ X:	97.778 ►
◄ Y:	102.991 ►
◄ Z:	20.466 ►
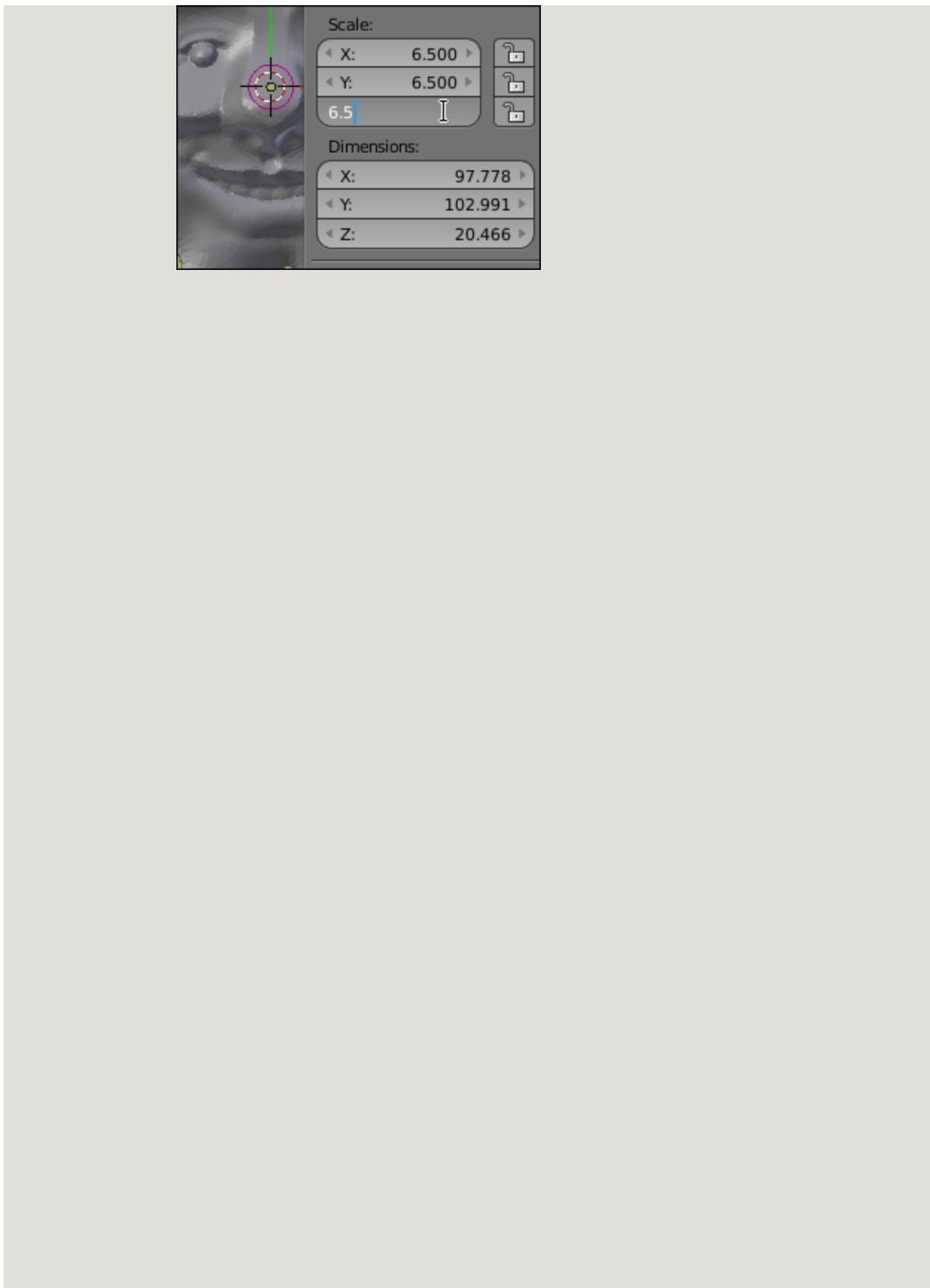
Building a puzzle piece

Now, let's build a basic puzzle piece that can be used to turn the sun into a puzzle.

Building the basic shape

Let's get started with the basic shape of a puzzle piece:

1. On the menu at the bottom of the **3D View**, locate the layers pane, which looks like an array of connected boxes. Each box is a separate layer.
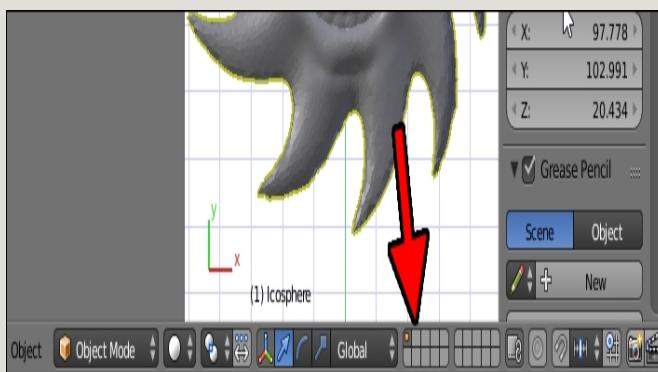
2. Click on an empty layer:
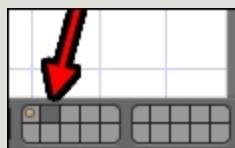
The sun will disappear, but that's simply because it's on another layer, the one with the orange dot in it. Organizing objects in different layers in Blender is a great way to ensure that things don't quickly get out of hand and to ensure that there is no need to hide or unhide individual objects. In addition, multiple layers can be viewed at once if desired, so you can work on related objects. For

now, we'll use this empty layer to start building our puzzle object:

1. Add (*Shift + A*) a **Grid** object:
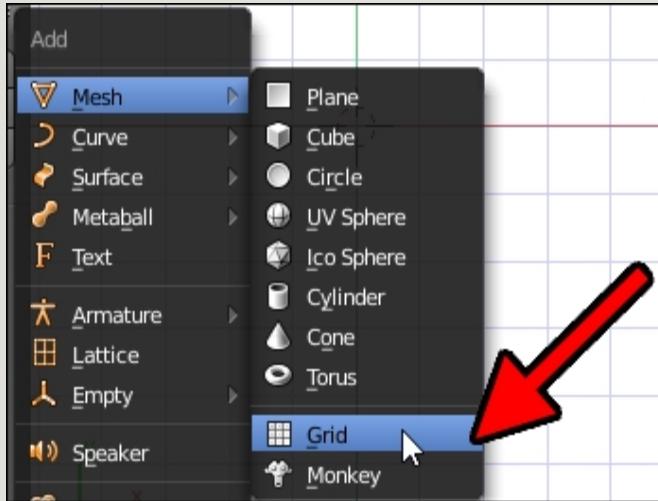
2. Under the **Tool Shelf** to the left of the **3D View** are the properties of the new **Grid**. Before clicking on anything else, change the **X Subdivisions** and **Y Subdivisions** values to 6:
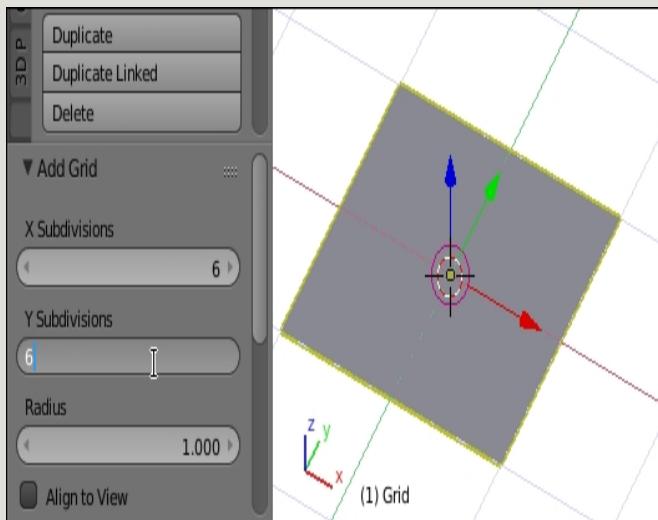
The **Grid** subdivisions won't be visible until you check its geometry by toggling **Edit Mode** quickly. For later steps, though, this piece needs to be positioned in the back-right quadrant of the 3D grid that we're working on. In other words, one corner of this grid needs to be sitting on the origin.

TIP

3. Move (*G*) the grid along the X axis (*X*) 1 unit.
4. Then, move (*G*) the grid along the Y axis (*Y*) 1 unit.
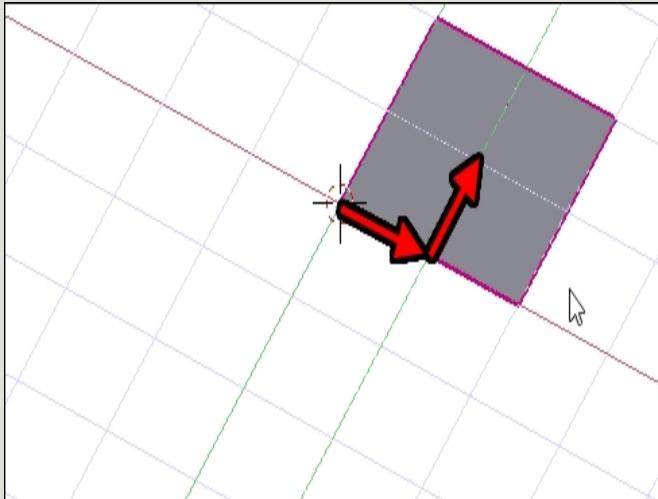
5. Enter **Edit Mode** (*Tab*) and deselect all points (*A*):
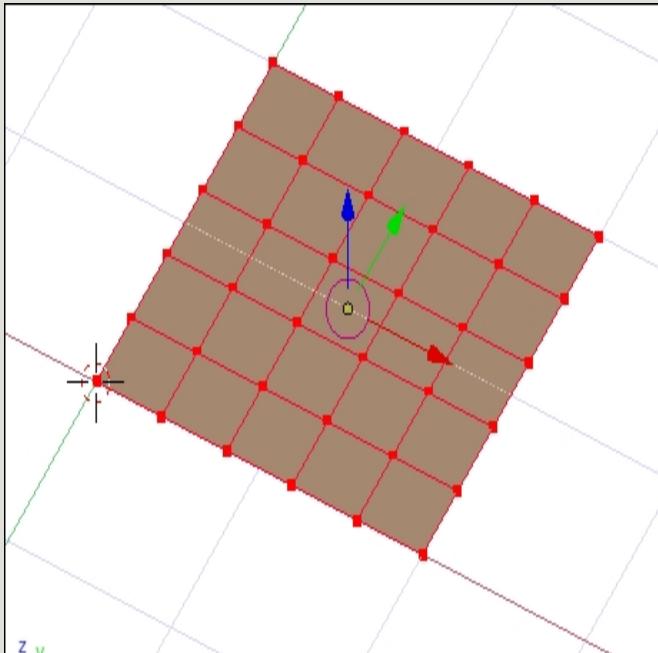
6. From the origin, select the second and third vertices to
   the right:
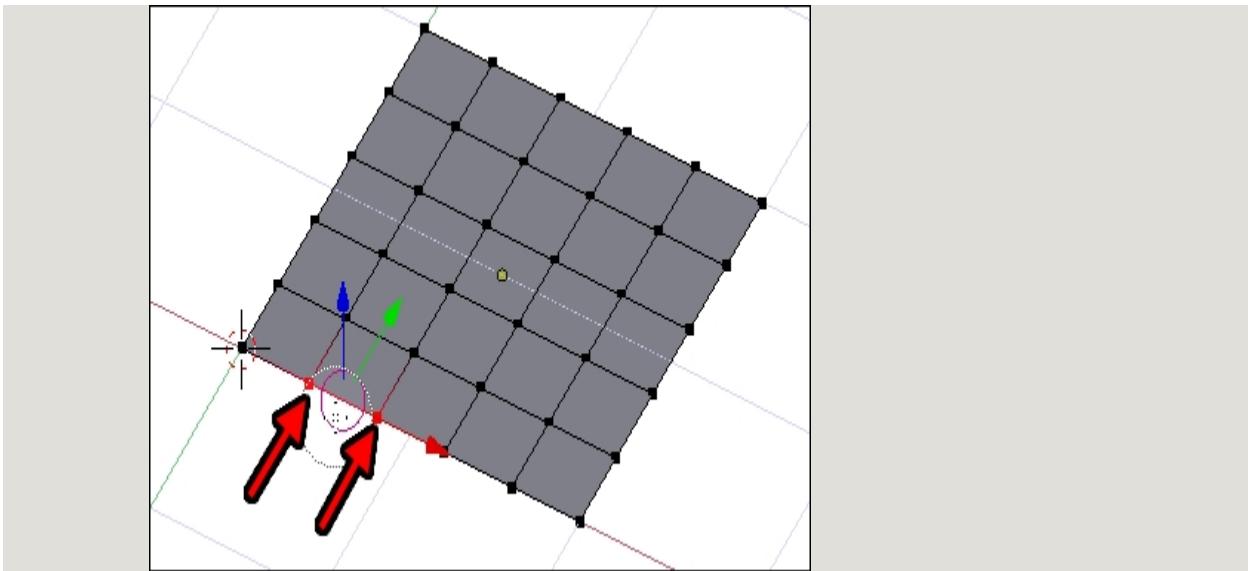
7. Delete (*X*) the selected **Edges**:
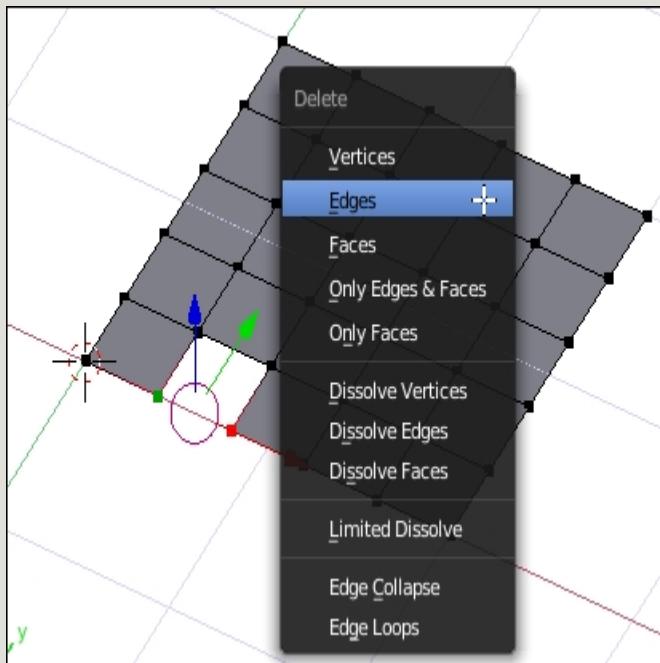
8. From the origin, select the second and third vertices in the Y direction:
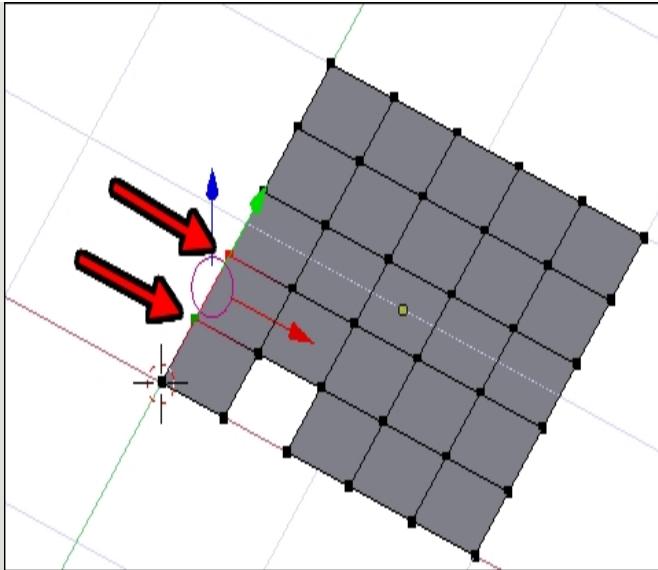
9. Extrude (*E*) the selected edge - 0.4 units. Be sure to type the amount to extrude on the keyboard, remembering to hit the - key when you do.
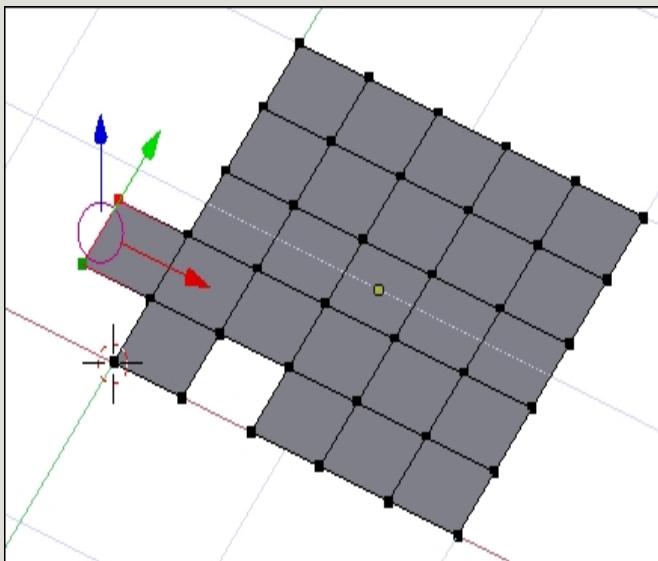
10. Scale (*S*) the selected edge by a factor of 1.5. Again, be sure to type the scale factor in.
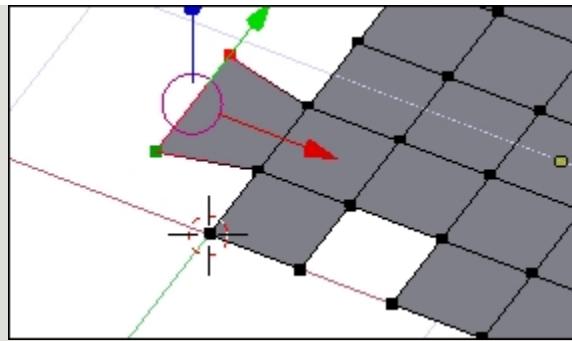
11. Select the two vertices at the top of the hole made when deleting the edge earlier:
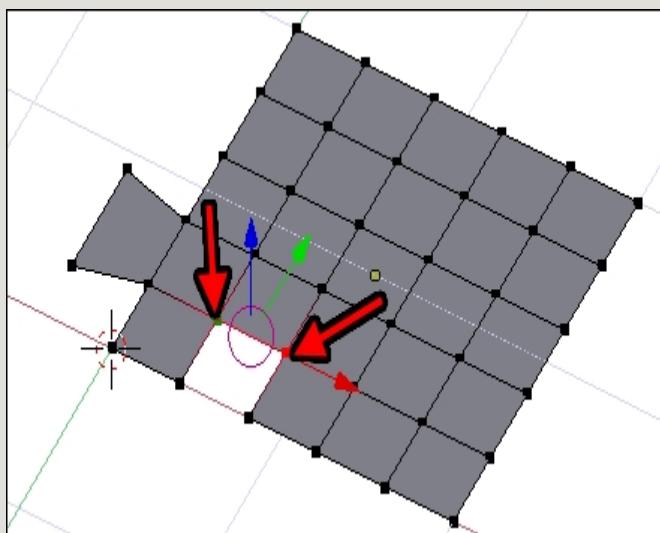
12. Scale (*S*) this edge by a factor of 1.5 as well:
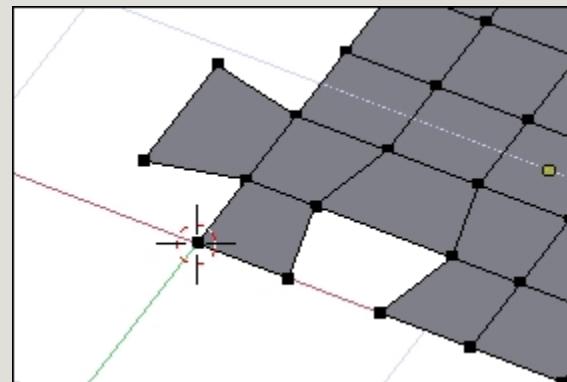
13. Leave **Edit Mode** (*Tab*).

This creates the basic, but flat, shape of an interlocking puzzle piece. Because both the tab and hole are made with complementary actions, they will mate perfectly.

Sizing the puzzle piece blank

Now to make the puzzle piece the appropriate size for the sculpted part:

1. In the menu at the bottom of the **3D View**, locate the **Pivot Center** drop-down menu, which looks like two overlapping circles by default:
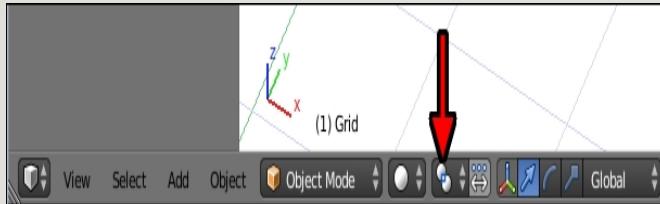
2. Click on the **Pivot Center** dropdown and select **3D Cursor**:
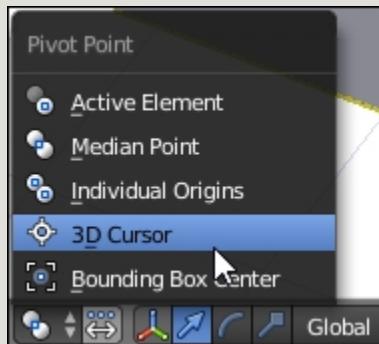
3. Be sure the **3D Cursor** is still at the origin by re-centering the view (*Shift + C*) if necessary.

> **NOTE**
>
> Changing the **Pivot Center** allows control of the point that the scale and rotation actions occur around. In this case, because the puzzle shape has been moved so that its corner is at the origin, the scale operation will keep that corner where it is as the rest of the piece grows. Notice how this works in the very next step.

4. Scale (*S*) the puzzle shape up some to make the next step easier.
5. With the puzzle shape still selected, hold down *Shift* and click on the layer that has the sun model in it:
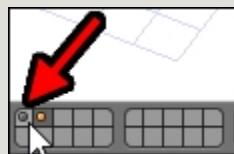
6. Scale (*S*) the puzzle up until it is bigger than the sculpted shape, along the X and Y axes:
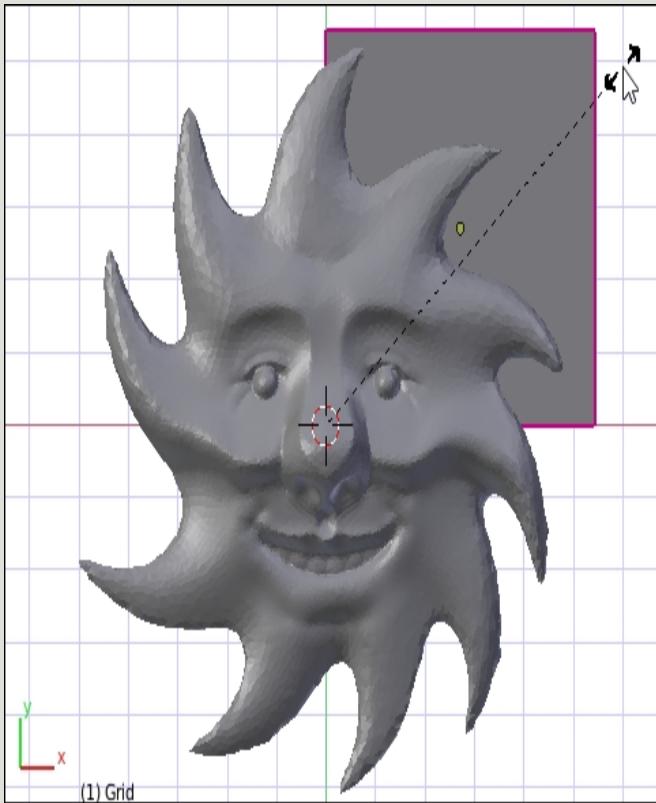
Adjusting the view and switching to **Wireframe view** (*Z*) can help with this step. The puzzle piece should completely cover a quarter of the sun model with the tab and blank completely inside the face without getting too close to the edge so that it can be a solid connection. If the rays of the sun reach too far to make this possible, the puzzle piece may have to be edited in **Edit Mode** to extend the edges along the X and Y axes until it fits.

7. Apply (*Ctrl + A*) the scale operation.

It is important to apply the scale operation because the modifiers we'll be using work on the object before scaling, and this project requires some precise control on the final model.

Turning a shape into an object

Let's get started with the shaping of the piece into an object:

1. With the puzzle shape selected, in the **Properties Panel**, locate and click on the **Modifiers** tab (the one that looks

like a wrench):
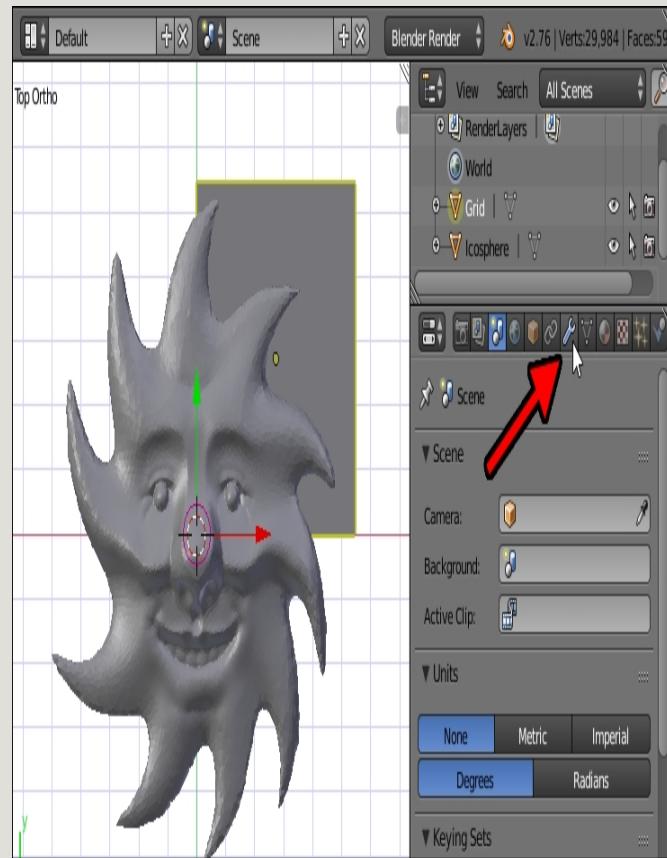
2. Click on the **Add Modifier** button and select the **Solidify** modifier:
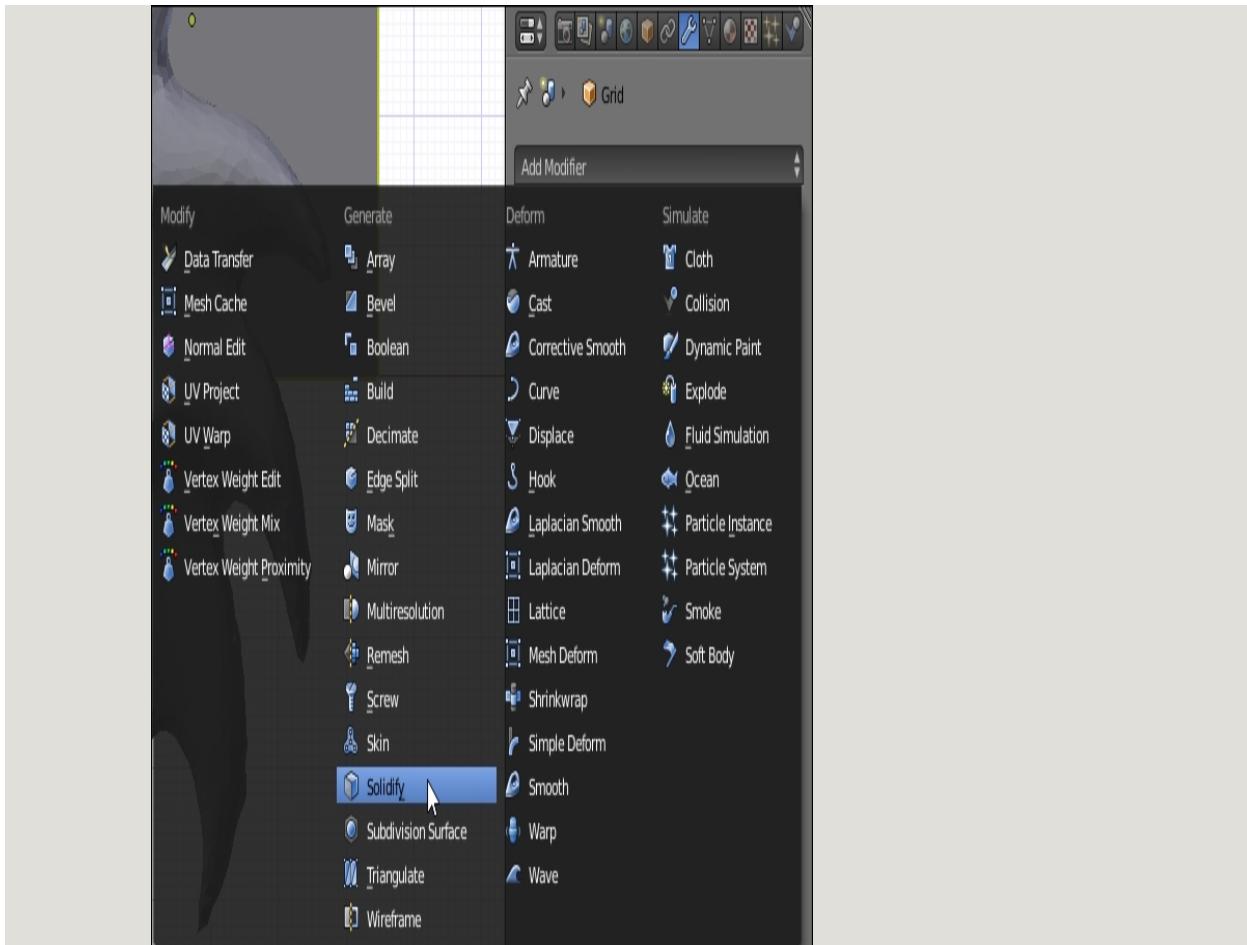
3. From the **Solidify** modifier properties, change the **Offset** value to 1.

> **NOTE**
>
> The **Offset** property tells the **Solidify** modifier where the newly solid shape should be created relative to the original flat shape. An offset of 0 creates a solid shape with the original shape in the exact middle. An offset of - 1 makes the original shape the top of the solid shape, relative to the normal of the flat shape. An offset of 1 makes the original flat shape the bottom of the solidified shape.

4. Change the **Thickness** property to something high enough that makes the puzzle piece thicker than the sun model:
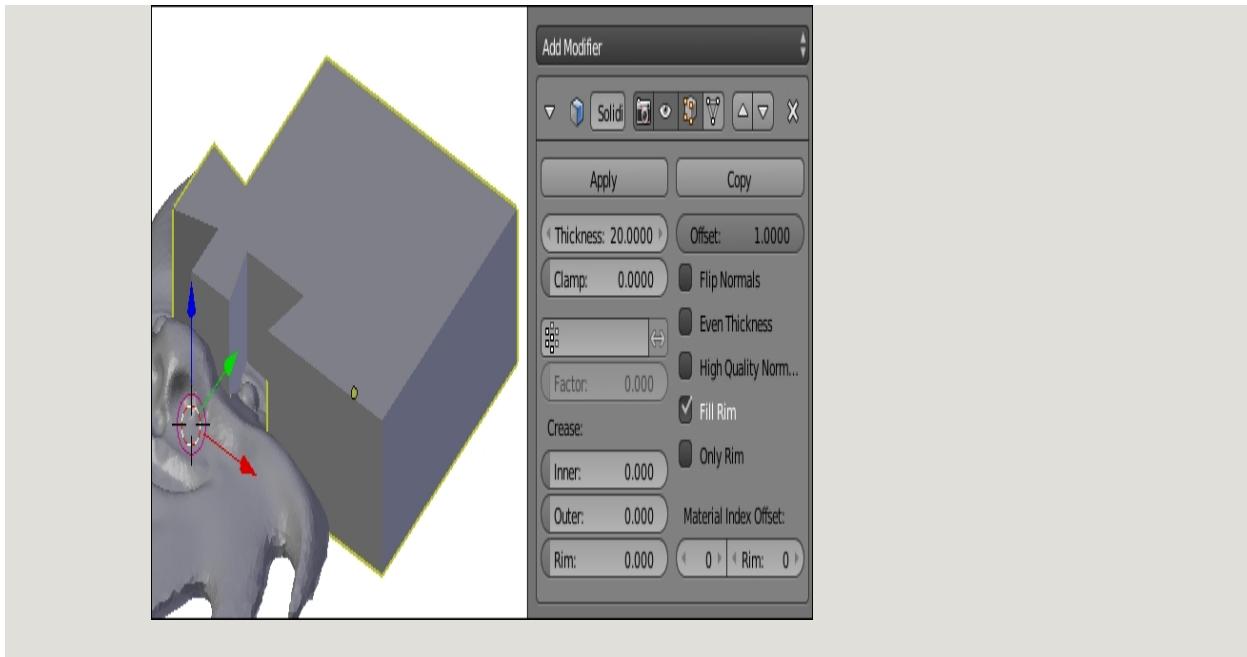
If the view is adjusted to look up from below, it can be seen that the puzzle shape, now a puzzle blank, doesn't extend to the bottom of the model. This is perfect because 3D prints need a flat bottom, and the puzzle blank will define a perfectly flat bottom, essentially cutting the bottom flat. This is nothing to worry about.

Adding some tolerance

When trying to make two objects that will interact with each other, such as the pieces of a puzzle, having pieces that are mathematically perfectly aligned is actually a bad thing. Edges in the design software that overlap perfectly will be too tight when made real to be able to slide past each other. So, the puzzle piece needs to be shrunk, just a little bit, to make it a suitable puzzle piece. Here's how you do it:

1. From the **Solidify** modifier, click on the **Apply** button:
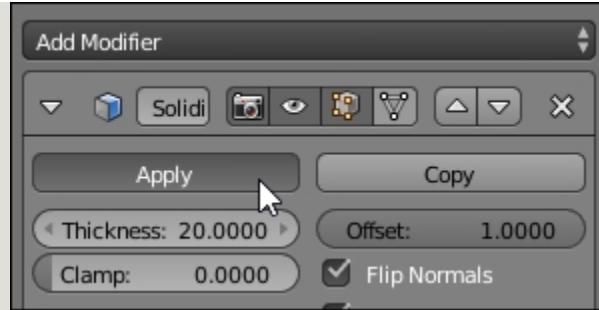
2. Click on the **Add Modifier** button and again select the **Solidify** modifier. On a flat object, the **Solidify** modifier turns it into a solid. On a 3D object, the **Solidify** modifier can be used to create a shell of controlled thickness.
3. Make sure the **Offset** value is - 1.
4. Adjust the **Thickness** value to 0.2.
5. Select the checkbox options for **Even Thickness**, **High Quality Normals**, and **Flip Normals** (leave **Fill Rim** checked):
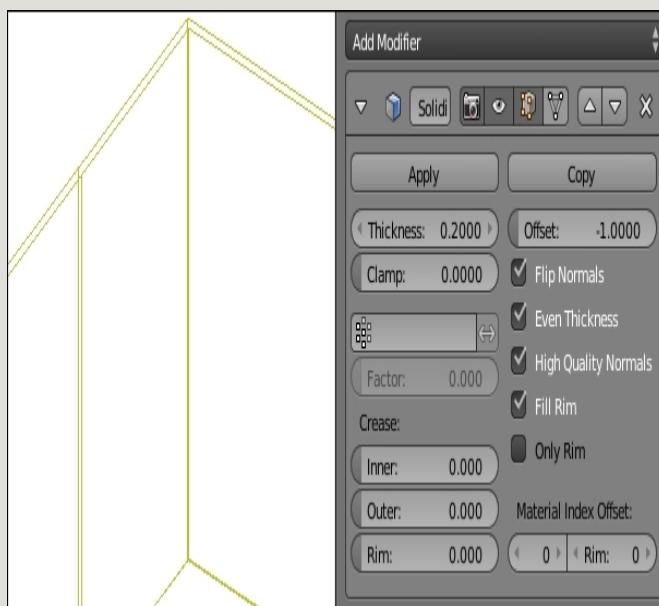
Besides the model getting darker in the **3D View**, there's no real indication of what the **Solidify** modifier did. Why did the model get darker? It is because its outside shell's normals are now pointing in the wrong direction as the **Flip Normals** option was checked. That's just a fancy way of saying that mathematically, the shape is now inside out, but that's okay because the inside-out shape is eventually going to be thrown away, with the inner shell being kept.

In **Wireframe view** (*Z*), on zooming in a bit on the edge of the piece, it can be seen that inside the puzzle piece is what looks like another, slightly smaller puzzle piece.

The next step is to get rid of the old, inside-out shell and keep just the slightly smaller puzzle piece.

6. Apply the **Solidify** modifier.
7. Enter **Edit Mode** (*Tab*).
8. In the menu at the bottom of the **3D View**, go to **Mesh | Vertices | Separate | By loose parts**, or press *P* on the keyboard and choose **By loose parts** in the menu that pops up:
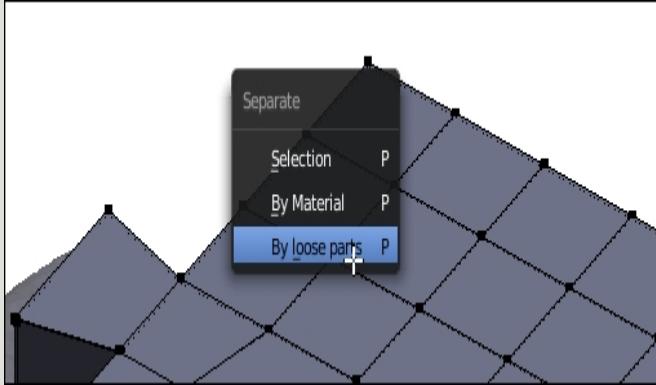
9. Exit **Edit Mode** (*Tab*).

There are now two puzzle shapes, one slightly smaller than the other. This can be confirmed by looking at the outliner, like this:

1. Deselect everything (*A*) and select just the outside puzzle blank.
2. Delete (*X*) the outside puzzle blank, leaving just the smaller inside one. You should see something like this:
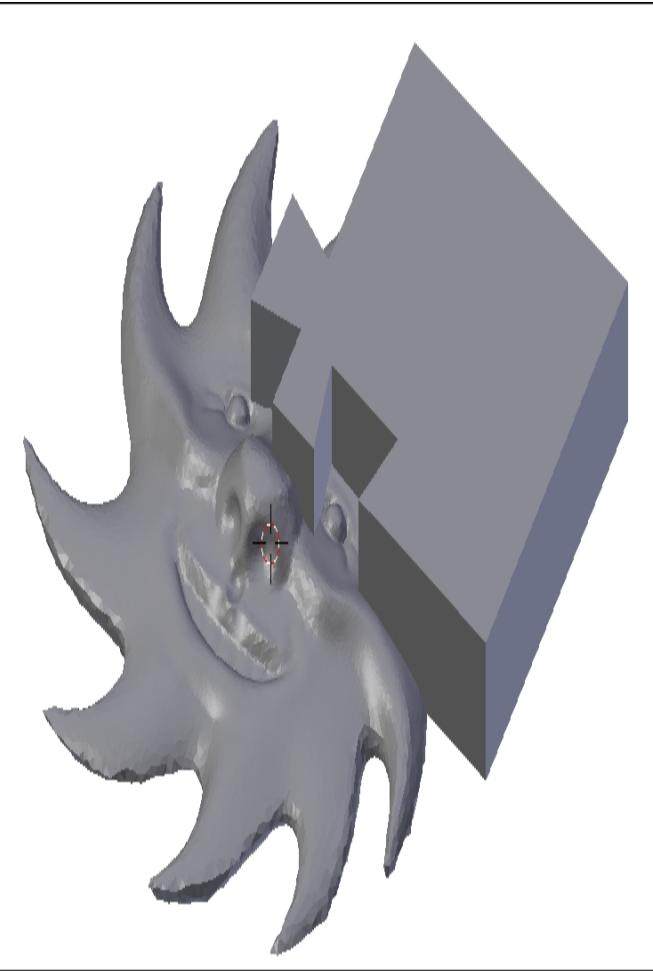
Putting it all together

We're nearing the end now. The puzzle blank has been built and been shrunk a little bit so that it works in the real world. Now to shape it like the sun model and make three more. Fortunately, this is going to be very easy:

1. Select the puzzle blank.
2. From the **Modifier** tab, click on the **Add Modifier** button.
3. Choose the **Boolean** modifier:
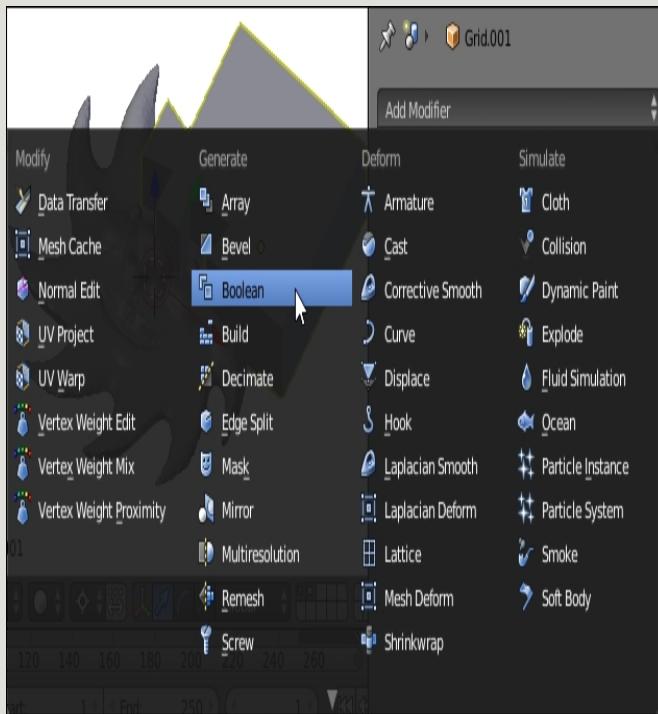
4. Be sure that the **Operation** option is set to **Intersect**.
5. Click on the **Object** dropdown and select the **Icosphere** object:

After a moment, the puzzle piece should effectively disappear. Where did it go? It's still there, but it's now shaped exactly like the part of the sun face it covered.

6. In the menu at the bottom of the **3D View**, click on just the second layer button to see the piece by itself:
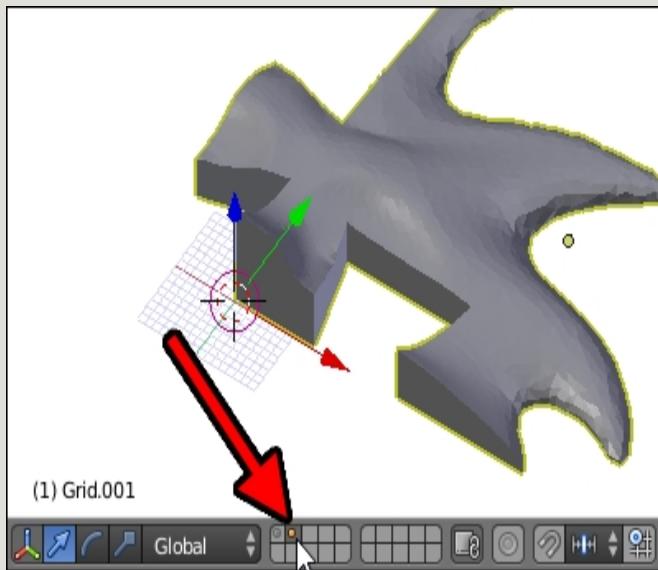
See? There it is—perfectly printable flat bottom and everything. Now, to make the other three pieces, continue with the following steps:

1. Click on the eye button in the **Boolean** modifier to temporarily disable it:

2. Duplicate (*Shift + D*) the puzzle piece and **Rotate** (*R*) it around the Z axis (*Z*) 90 degrees:
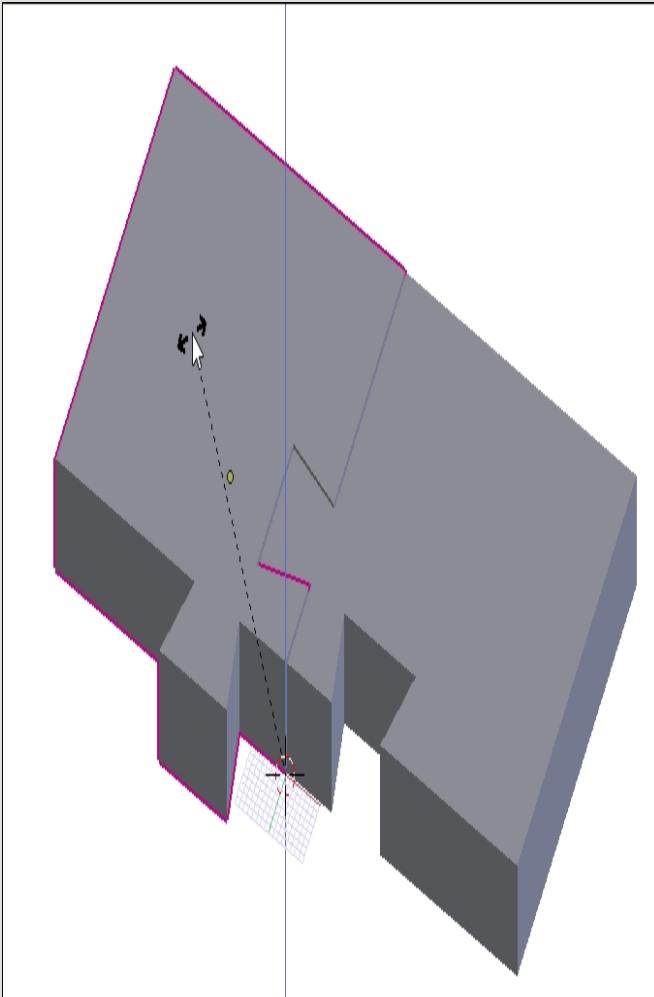
Because the **Pivot Center** was switched to the **3D Cursor** earlier, the rotation happens around the middle. Also, because the pieces were designed well, they should interlock perfectly with just a little gap between them.

If the rotation is askew, it may be because the **3D Cursor** has moved. Undo (*Ctrl + Z*) the duplication, recenter the **3D Cursor** (*Shift + C*) and try again.

3. Repeat duplicating and rotating the puzzle blank two more times:
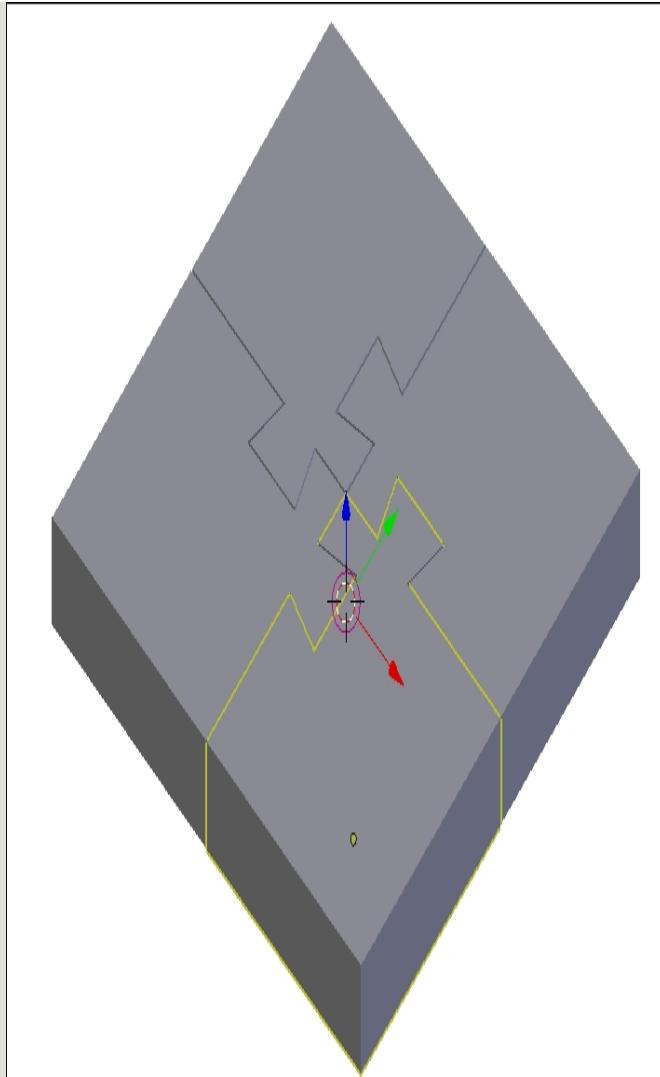
4. For each puzzle blank, click on the eye button in the
   Boolean modifier to re-enable it:
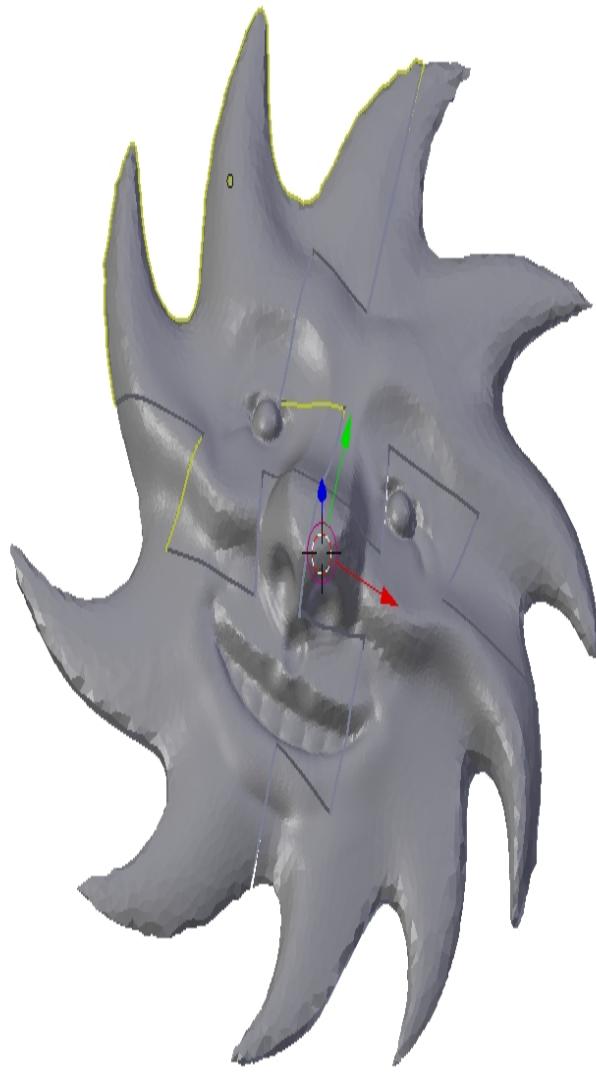
What if the Boolean modifier doesn't work?

In Blender, the complex math behind the **Boolean** modifier sometimes fails to work properly. Sometimes, it's because there are errors in the geometry of either object or the geometry is too complex. A sculpted object has very complex geometry. So, it's possible that the **Boolean** modifier doesn't perform as expected. If that happens, here's a list of things to try to fix it:

1. Apply (*Ctrl + A*) the rotation and scaling of both objects.

2. In **Edit Mode**, select all vertices (*A*) and **Recalculate Normals** (*Ctrl + N*).
3. Use the **Decimate** modifier to simplify the geometry of a complex part—in this case, the sun model. Add a **Decimate** modifier and adjust the **Ratio** down as far as it will go before a noticeable difference in the geometry of the model is seen.
4. Slightly modify the geometry of the shapes. In this case, try editing the piece blank in **Edit Mode** (*Tab*). Select all the vertices on the bottom of the puzzle piece with a **Box Select** (*B*), and **Scale** (*S*) them along the Z axis (*Z*) by 0 to reflatten the bottom. If this works, remember to edit the other pieces so that they all match.

Usually, one of these actions, or a combination of them, will kick the **Boolean** modifier into working, and your model will be complete.

Exporting and printing

Now that the pieces are all properly shaped,
select each piece and, in the **Info** panel at the
top of the screen, go to **File** | **Export** | **STL**. Give
each piece a unique name, and they'll be ready
for 3D printing and playing with:
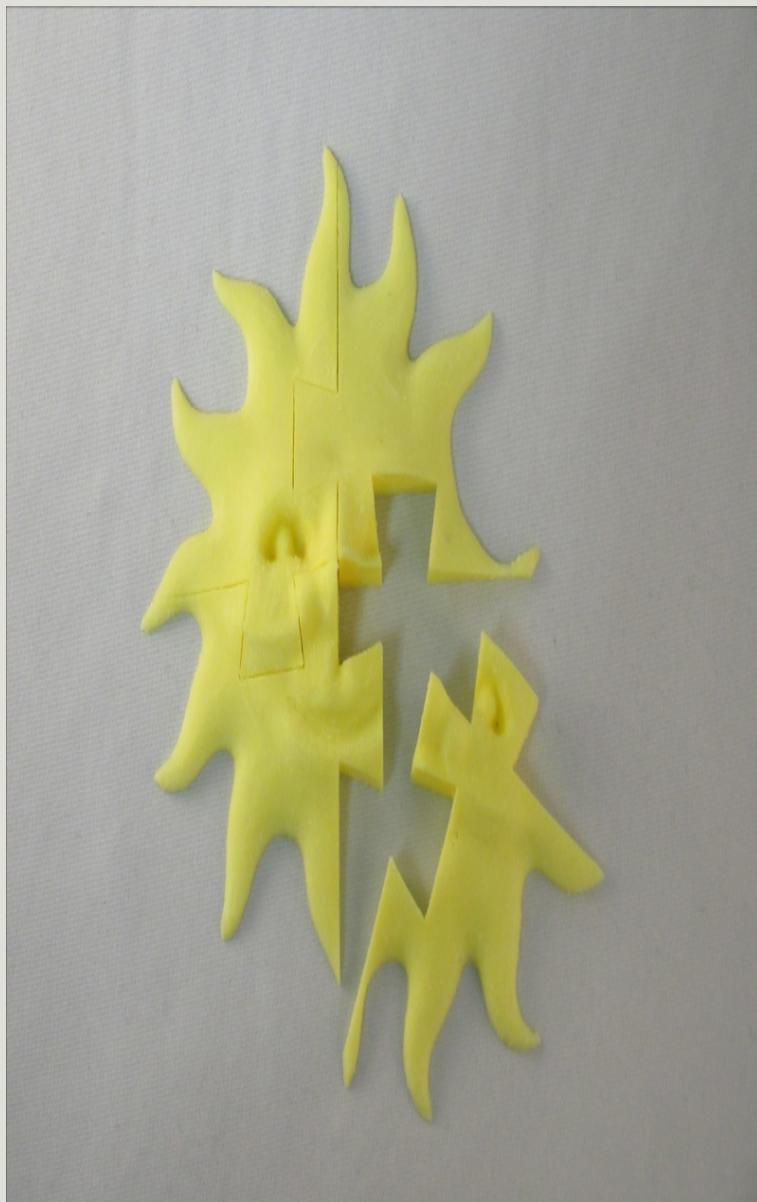
Summary

In your grandparents' or parents' garage, they might have had a jigsaw and tools that they made cute wooden puzzles with—clunky shapes that vaguely resembled an elephant or truck. But nowadays, we can do so much more with 3D printing. It's not difficult to see how this project can be applied to a puzzle of any shape, just as complex or robust as a loved one's favorite thing:
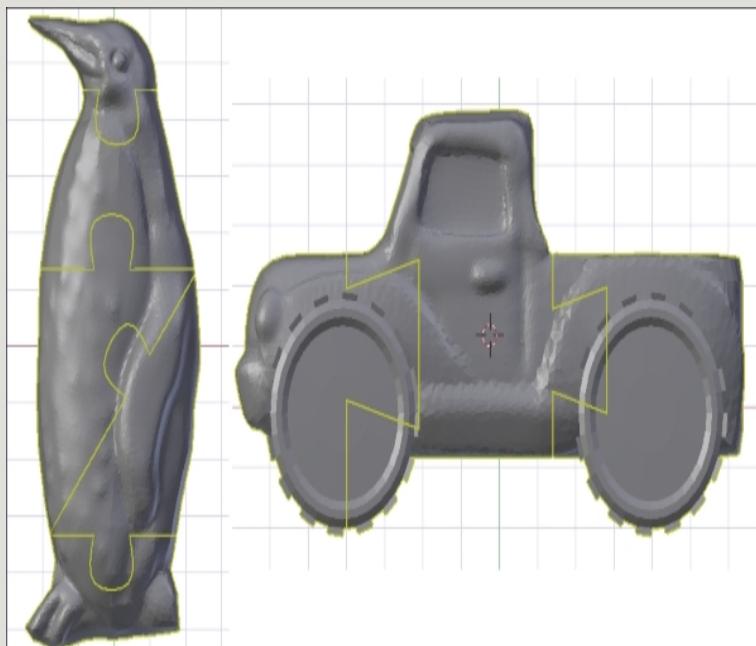
Blender's unique collection of tools seems eclectic to some, but in the end, they enable robust interactions that result in us being able to make projects such as this puzzle without having to switch tools to finish. Models can be sculpted and then, without switching tools, combined with rigid shapes to create complex but useful and fun objects for 3D printing. This project also taught you about tolerances in 3D-printed parts and how they can be adjusted to make parts that interact with each other. While different 3D

printers may require different tolerances, the skills learned in this chapter can be applied to adjust those tolerances.

Index

A

- additive manufacturing

    - about / What defines 3D printing?

- axis locking

    - about / Controlling transformations

B

- Blender

    - using / Why Blender?, The price is right

    - downloading / Downloading and installing Blender

    - installing / Downloading and installing Blender

    - URL / Downloading and installing Blender

    - image, importing into / Importing the image into Blender

    - references / Importing the image into Blender

    - reference pictures, increasing / Increasing reference pictures

- Blender units

    - defining / Blender to real life

- bottom, pencil holder

    - flattening / Flattening the bottom

C

- calipers

    - using / Measuring with calipers

    - about / Measuring with calipers

    - manual caliper / Manual or Vernier calipers

    - Vernier caliper / Manual or Vernier calipers

    - digital calipers / Digital calipers

- computerized numerical control (CNC)

    - about / What defines 3D printing?

D